UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE INFORMÁTICA

THREAT INTELLIGENCE:

USING OSINT AND SECURITY METRICS TO ENHANCE SIEM CAPABILITIES

João Paulo Martins José Teixeira Alves

MESTRADO EM SEGURANÇA INFORMÁTICA

Dissertação orientada por:
Profª. Doutora Ana Luísa do Carmo Correia Respício
e Eng. Pedro da Silva Dias Rodrigues

2017

Agradecimentos

Antes de expressar a minha gratidão a quem ajudou-me nesta fase da minha etapa académica, quero salientar que caso não sejam mencionados neste documento, é apenas esquecimento da minha parte e quero que saibam que estou inteiramente grato. Quem me conhece sabe que não sou o tipo de pessoa que demonstra a sua gratidão por palavras, mas sim por acções.

Os meus agradecimentos ao projecto DiSIEM, financiado pela Comissão Europeia no programa H2020, G.A. nº 700692, por ter financiado este projecto.

As primeiras pessoas que quero agradecer são aos meus orientadores, Prof. Doutora Ana Respicio e Eng. Pedro Rodrigues, não só por me terem aceite como seu orientando e me terem proporcionado esta experiência, como todo o seu apoio, ensino e total disponibilidade sempre que foi necessário, o meu obrigado, pois aprendi muito com ambos. Agradeço a Ivo Rosa e Gonçalo Martins da EDP. Ivo ensinaste-me bastante e ajudaste sempre que pudeste, agradeço os nossos debates semanais sobre novas ideias e que potenciais cursos este trabalho podia seguir, já o disse e volto a dizer tu és uma fonte de ideias e só é preciso estar alguém atento para escutar e pôr em prática, eu tentei ser essa pessoa. Não foste meu orientador, no entanto, houve momentos que me orientaste, muito obrigado! Gonçalo foste a pessoa que me fez sentir logo como membro da equipa. Estiveste sempre disponível, mesmo com imenso trabalho, sempre ajudaste-me. Tiveste sempre paciência para mim, não sei como, até para responder às mil e uma perguntas diárias que tinha. Ensinaste-me não só o essencial para o meu trabalho, mas muito mais do que era necessário, e sei que essa aprendizagem vai ser útil no meu futuro. Estiveste sempre atento e preocupado com o meu trabalho e zelaste por mim, obrigando-me a fazer as pausas para descansar e almoçar para repor a energia para voltar ao trabalho. Estou grato pela tua constante preocupação e ajuda.

Agradeço à equipa do SOC (incluindo o estagiário) por ajudarem-me a monitorizar os incidentes para a tese e a regra criada. Sempre preocupados com o meu trabalho e se eu precisava de algo. Fizeram-me sentir como membro da equipa, sei que vocês aprenderam comigo, mas eu aprendi muito com vocês e deram alegria ao trabalho, mesmo em momentos de aflição. Não quero esquecer de todos os outros membros da equipa DSI da EDP (incluindo o colega do Brasil e parceiro de secretária), agradeço-vos a todos pela

ajuda, bem estar e momentos divertidos que passei enquanto estive com vocês. Sei que posso contar com vocês no futuro e por essa confiança agradeço.

Agradeço aos membros do projecto pela ajuda que me deram nas reuniões, ao debatermos sobre novas ideias para melhorar o trabalho.

A todos os membros da sala 6.3.34, não quero esquecer-me de ninguém, vocês sabem quem são, agradeço a todos pela companhia, momentos bem passados, onde havia tempo para risadas, discussões "civilizadas" e tempo para ajuda mútua no trabalho. Sei que fiz novas amizades e as antigas apenas fortaleci. Também sei que estas amizades não ficarão apenas na sala.

Por último agradeço aos meus pais, avós, resto dos familiares e amigos por estarem sempre a refilar comigo e a tentar combinar jantares para descontrair depois de um dia de trabalho, vocês são maçadores, mas boas pessoas, preocupados sempre com o meu estar e com o estado do meu trabalho, agradeço-vos imenso.

This work is supported by the European Commission through the H2020 programme under grant agreement 700692 (DiSIEM).

Dedicado a familiares e amigos

Resumo

Nos últimos anos, face ao aumento em quantidade e em complexidade de ataques informáticos contra diversas organizações, tem-se verificado um crescimento elevado no investimento em plataformas de segurança informática nas infra-estruturas das organizações. As equipas com a responsabilidade de garantir a cibersegurança necessitam de monitorizar um vasto número de dispositivos, utilizadores, aplicações e, consequentemente, eventos de cibersegurança relacionados com esses elementos. A plataforma mais utilizada para monitorizar os eventos de segurança informática é o sistema de Gestão e Correlação de Eventos de Segurança (SIEM, do inglês *Security Information and Event Management*). Este sistema agrega toda a informação de segurança proveniente de diversas fontes, normaliza-a, enriquece-a e envia-a para uma consola centralizada de gestão. A eficiência e a eficácia das equipas de resposta a incidentes de segurança dependem em grande medida da capacidade de o sistema produzir uma alarmística detalhada e contextualizada sobre possíveis ameaças. Para melhorar essa capacidade é necessário conjugar indicadores externos relevantes com a informação recolhida na infra-estrutura da organização.

Threat Intelligence (TI) é o conhecimento adquirido da conjugação das técnicas de recolha de informação sobre ameaças externas à organização e das técnicas de recolha de informação sobre factores de segurança internos das organizações. É necessário estar atento às fontes públicas de informação de cibersegurança e avaliar a sua qualidade para obter indicadores fidedignos sobre actividades maliciosas.

A organização necessita de avaliar o seu nível de cibersegurança para identificar as vulnerabilidades existentes, antes que estas possam ser exploradas por agentes mal-intencionados. Somente com o recurso a fontes de informação, internas e externas, é possível ter uma abordagem TI abrangente e aplicar as medidas de cibersegurança adequadas para evitar os ciberataques aos quais a organização possa estar vulnerável.

Para uma organização estabelecer correctamente o seu nível de cibersegurança, é necessário realizar uma gestão de risco adequada. A gestão de risco é caracterizada por três etapas, todas interligadas e contínuas: análise do risco, avaliação do risco e controlo do risco. No fim do processo, a organização terá um conhecimento credível sobre o seu risco informático, tendo um bom suporte para as tomadas de decisão no que respeita a reestruturações e investimentos em segurança informática.

As métricas de segurança são a ferramenta mais indicada para o processo de gestão de risco. Estas ajudam a determinar o estado de cibersegurança no qual a organização se encontra, o desempenho da equipa do Centro de Operações de Segurança (SOC, do inglês *Security Operation Center*), e o nível de segurança das infra-estruturas da organização. As entidades governamentais e militares foram as primeiras a utilizar as métricas de segurança. No entanto, recentemente, investigadores de diversos tipos de organizações (públicas, privadas e público-privadas), têm investido recursos para melhorar e implementar estas métricas nas suas organizações. Toda esta atenção dada às métricas de segurança deve-se ao resultado evidente da sua implementação: é possível medir o risco, classificá-lo e, finalmente, tomar as contramedidas adequadas para reduzir o impacto de possíveis ciberataques, aumentando a cibersegurança na organização. Contudo é necessário estabelecer os objectivos e o propósito das métricas de segurança. Muitas equipas de cibersegurança cometem o erro de criar métricas que são complexas, fora do contexto, e expressam resultados com valores irrealistas. O resultado desta má gestão das métricas de segurança é oposto do pretendido, providenciando má informação e, consequentemente, diminuindo a cibersegurança de uma organização. A visualização dos resultados das métricas é o último passo da criação de métricas e tem como finalidade fornecer informação de uma forma ilustrativa, com recurso a formatos de fácil leitura e compreensão. As visualizações ajudam a equipa responsável pela cibersegurança de uma organização a visualizar de imediato informações relativas ao nível de cibersegurança dos sistemas e o risco de cada activo. As visualizações permitem à equipa avaliar e responder, de uma forma quantitativa e qualitativa, às perguntas colocadas pela direcção executiva, tais como: qual o nível de segurança, qual o valor de risco na organização, qual o retorno financeiro dos investimentos feitos para melhorar a segurança informática na organização ou mesmo para justificar a permanência, redução ou aumento de equipamentos e equipas de cibersegurança.

Para além do mecanismo de descoberta de informação interna, o *Open Source Intelligence* (OSINT) é considerado o mecanismo para a captura de informação externa a partir de fontes online. Com um conjunto de técnicas é possível capturar a informação relevante para o conhecimento sobre ciberameaças. Existem comunidades de cibersegurança cujo objectivo é publicar listas com informações sobre novos ciberataques, que normalmente contêm informações sobre anfitriões suspeitos ou conteúdos maliciosos. Estas listas, as listas negras, podem ser públicas, quando qualquer pessoa pode aceder à sua informação, ou privadas, restringindo o uso das listas a um determinado grupo ou comunidade. Apesar de as listas oferecerem uma informação valiosa sobre ciberameaças actuais, estas sem qualquer tipo de pré-processamento, podem gerar um número significativo de falsos positivos, devido à ausência de contextualização e alinhamento com a realidade da organização.

Este trabalho é dividido por dois tópicos: métricas de segurança e listas negras confiáveis. Para cada tópico são descritas soluções para melhorar o estado de segurança numa organiza-

ção, ao integrar o processo TI em tempo-real no SIEM. Esta integração pode ser materializada na utilização de métricas de segurança para análise do estado de segurança na organização e fontes de segurança com informação sobre endereços IP suspeitos de actividades maliciosas com consideração das operações da equipa do SOC sobre incidentes de segurança, com o recurso a métricas. A utilização directa das listas negras, sem qualquer tipo de pré-processamento, resulta num elevado número de falsos positivos, pela ausência de contextualização e alinhamento com a realidade da organização.

O trabalho está inserido no projecto DiSIEM e resulta da colaboração de dois dos parceiros do projecto, Faculdade de Ciências da Universidade de Lisboa e EDP - Energias De Portugal, SA. Os objectivos alinham-se com as metas do projecto DISIEM: 1) fornecer informações OSINT para um sistema SIEM, melhorando a sua detecção e prevenção de novas ameaças; 2) identificar e desenvolver um conjunto de métricas dedicadas à equipa de cibersegurança para uma melhor gestão e monitorização dos eventos de segurança para aumentar o estado de segurança na organização, consequentemente, reduzindo o risco de actividades maliciosas na organização.

A dissertação apresenta e discute um conjunto de métricas com uma estrutura bem definida para serem aplicadas no sistema SIEM. Estas métricas cobrem os sectores de gestão, processos e tecnologia, e estão apropriadas para a realidade da equipa de cibersegurança. É introduzido protótipos para visualização dos resultados das métricas, incluindo dados históricos, possibilitando assim uma avaliação comparativa de eficiência.

O trabalho propõe uma solução OSINT para aperfeiçoar a alarmística do sistema SIEM, reduzindo a taxa de falsos positivos, com base na avaliação do nível de confiança em fontes de informação públicas, e dessa forma contribuir para a eficiência das equipas de cibersegurança nas organizações que usam o sistema SIEM. Esta solução usa listas negras que identificam endereços de Protocolo de Internet (IP do inglês *Internet Protocol*) suspeitos de actividade maliciosa. A informação pode ser sobre sua maliciosidade, o número de denúncias (efectuadas por comunidades ou outras listas negras), número de ataques aos quais o endereço IP esteve associado, a última vez que foi denunciado, entre outros. As listas negras são úteis para serem utilizadas no sistema SIEM, para a monitorização de comunicações entre a organização e um IP suspeito. Assim, quando houver um alarme de uma comunicação suspeita, a equipa do SOC pode actuar de forma imediata e analisar os eventos para identificar a máquina, pedir uma análise local e eliminar a ameaça, caso seja detectada.

A solução recolhe informação sobre endereços IP de um conjunto de listas públicas. Os endereços IP e as listas são avaliadas quanto à sua veracidade, com base na correlação da informação recolhida a partir das listas e com base em métricas sobre o resultado dos incidentes associados a comunicações suspeitas entre a organização e endereços IP das listas. Esta avaliação é realizada de forma constante, sempre que exista uma alteração nas listas públicas ou nos incidentes, para que os seus valores sejam os mais actualizados e

precisos.

Foi desenvolvida uma aplicação para administrar as listas negras utilizadas, os endereços IP, os casos da organização e endereços públicos da organização. São apresentadas regras do SIEM que seleccionam os endereços IP recolhidos das listas negras com base na reputação dada pela avaliação da sua veracidade, para a monitorização de comunicações entre a organização e os endereços IP suspeitos.

Os resultados mostram que há um aumento de detecção de casos positivos com a utilização da solução proposta. Este aumento deve-se ao uso de informação interna dos incidentes, tratados pela equipa do SOC, como parâmetros de avaliação da confiabilidade das listas negras e dos endereços IP. Dois componentes que se destacam como parâmetros de avaliação da confiabilidade é o componente da precisão e o componente da persistência. O componente da precisão tem em conta os resultados da organização e aumenta a confiabilidade de um endereço IP ou de uma lista caso o número de resultados positivos dos casos de incidentes relacionados com o IP seja superior ou número de resultados falsos positivos. A persistência tem em conta a precisão e a denúncia de um endereço IP por parte das listas, para o guardar na nossa lista durante três meses.

A avaliação da lista negra e do seu conteúdo considerando o ambiente da organização é uma solução que não foi apresentada por nenhum outro trabalho, e o mais semelhante é o uso de métricas ou recolha de informação com o uso do conceito OSINT, sem avaliação do conteúdo com base na informação da organização. Sendo um trabalho inovador, este ainda se encontra na sua fase primordial. Os resultados do nosso estudo servirão como base para melhorias e comparação de resultados de estudos posteriores para melhoria na avaliação da confiabilidade das listas públicas e da maliciosidade do seu conteúdo.

Palavras-chave: métricas de segurança, SIEM, OSINT, listas negras, internet protocol, ciberameaças, threat intelligence

Abstract

Threat Intelligence (TI) is a cyber defence process that combines the use of internal and external information discovery mechanisms. The Security Information and Event Management (SIEM) system is the tool typically used to aggregate data from multiple sources, normalize, enrich and send it to a centralized management console, later used by the security operation team (SOC). However, it is necessary to use Security Metrics (SM) to summarize, calculate and provide valuable information to the SOC team from the large datasets collected in the SIEM. Although the SM provide valuable information, its erroneous creation or use could lead to the opposite goal and decreasing the security level, by generating false positives.

Regarding the external information discovery, the information from blacklists is commonly used to monitor and/or to block external cyberthreats. The blacklists provide intelligence about suspicious Internet Protocol (IP) addresses, reported by communities and security organizations. Although the use of blacklists is commonly used to detect suspicious communications, it generates a high rate of false positives.

We introduce a set of security metrics, well-structured and properly defined to be used with a SIEM system. We develop a solution with Open-Source Intelligence (OSINT) mechanism to discover and collect suspicious IP from public blacklists, a process to assess the reputation of the suspicious IP addresses and blacklists, considering the persistence of the IP and the organization's incidents of communications with suspicious IP addresses. The IP are inserted in the SIEM with rules to monitor and aiming at reducing the number of false positives.

The preliminary study in a real environment shows that the proposed solution improves the security effectiveness of the SIEM's alerts due the innovations idea of assessing the IP and blacklists by using the persistence and precision components, and considering the organization's incidents status.

Keywords: security metrics, SIEM, OSINT, blacklist, internet protocol, cyberthreats, threat intelligence

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In recent years, due to the increase in the number and complexity of cyberattacks against organizations, there has been an increase in the investment in Information Technology (IT) security solutions in the organizations' infrastructures. Teams responsible for the organization's cybersecurity need to monitor a vast number of devices, users, applications and, consequently, cybersecurity events related to these elements. The typical platform used to monitor those events is the Security Information and Event Management (SIEM) system. This system aggregates all the information about cybersecurity events from various sources, normalizes it, enriches it and sends it to a centralized management console. The effectiveness of the cybersecurity incident response team depends on the capability of the SIEM to produce detailed and contextualized alarms for possible threats, and the use of SM that can evaluate the security degree of the organization and the performance of the SOC team. To improve this capacity, it is necessary to combine relevant external indicators with the information gathered in the organization's infrastructure, and structure SM that are suitable for the SOC capabilities.

Threat Intelligence (TI) is the process of extracting information about cyberthreats from diverse sources (internal and external). It is necessary to be aware of the Internet cybersecurity information sources, to obtain reliable indicators about cyberthreats - external source - and extract knowledge about the organization's security status, in order to identify the vulnerabilities that can be exploited by an attacker - internal source. Only with the combination of both sources, external and internal, it is possible to have a thorough TI approach and apply the security measures to reduce the risk of cyberattacks, thus enhancing the security status of the organization.

Security Metrics (SM) are used to assess the security status, the performance of the Security Operation Center (SOC) team, and the security and health of the infrastructures in the organization. In the areas of science, the term 'metric' is used over 200 years. Although in the decade of 1960, SM were already investigated and implemented by the government [38], only in recent years, they are getting more attention for improvements and implementations by researchers from all types of organizations (private, public, mil-

itary, and more). In addition to provide knowledge about the weakness and flaws within the organization (security status), the performance and work done by the information security team and cybersecurity appliances, SM also provide relevant indicators about malicious threats. SM prove that their usage significantly enhances the risk's measurement, thus providing information about the vulnerable assets, the dependencies between them, and the most critical sectors within the organization [23, 33, 39]. The C-level managers, can use this information to make well-supported decisions in cybersecurity strategies for counter measures to reduce the impact of cyberthreats. Therefore, SM can enhance the organization security status [23, 33, 39].

If the SM are used for the internal source then the sources commonly used to retrieve external information about the current existing cyberthreats are known as cybersecurity feeds, especially the blacklists. Blacklists are lists containing information about suspicious hosts or malicious contents. This work uses blacklists that identify Internet Protocol (IP) addresses. The lists can be public, i.e. anyone can retrieve information from them or private, restrict lists to be used by a particular group or community. The information can be about their maliciousness (botnet, phishing, ransomware, or DoS), the number of reports (by user's communities or other blacklists), number of attacks, last time reported, and more. Although the typical approach is to block the communications with these suspicious IP addresses, this approach does not consider the probability of a machine already being infected, and only prevents the malicious communication. The blacklists are helpful for the SOC team to monitor communications between the organization and a suspicious IP. When there is an alarm, the Security Operation Center team (SOC) can take immediate action and analyse the asset to detect and eliminate the infections.

This work is part of the Diversity enhancements for Security Information and Event Management (DiSIEM) project [12] and was implemented in collaboration with two of the organizations that form the consortium: Faculty of Science of the University of Lisbon and EDP - Energias de Portugal, SA.

"If you know the enemy and know yourself, you need not fear the result of a hundred battles.", [40]

1.1 Motivation

As the cyber information security team is getting more assets to manage and secure, arises the need to create Security Metrics to measure all the security environment. These SM should cover all the levels, starting from the technical/operational view and reaching to the C-level security manager. There are fundamental questions that should be questioned and answered when creating and implementing security metrics, such as which SM should be used, what is the proper raw data to feed the SM, what to extract and display from the SM

and how to incorporate with the collector system current in use. Although SIEM systems are built with predefined SM, it is crucial for the safety and security of the organization to implement into the SIEM system or with information provided by it, custom metrics tailored to the environment, context and objectives of the organization.

The Internet is a vast source of information and can provide knowledge to the organization about cyberthreats. But how to use it, where to use it, what to do with that information, how to implement in a manner without compromised or modifying completely the workflow and technologies used (such as the SIEM system), are issues to be resolved. And how trustworthy is the gathered information? In addition to gather the information is necessary to classify their trustworthiness and reliability, accordingly to the organization's environment and security status.

Due to the potential valuable knowledge of cyberthreats which the Internet and the share information between organizations can offer, the information security teams already use private lists. Although the public lists produce a significant number of false positives, the teams are starting to use it, yet only use the public lists that have a certain level of trust. This level of trust is from the team's experience in using the lists and the lists reputation in the cybersecurity communities.

As referred in the related work, no work was found about classifying a set of public lists and their content with insight information about the organization's security status and applying that knowledge to calculate the trustworthiness of the credibility of a blacklist and trustworthiness of the suspicious maliciousness of an IP address. The organization should have, on their side, a method or a system capable of receiving information from multiple sources and classify them accordingly with the correlation between those sources and the environment of the organization, providing an output list with a reputation score, more accurate, reliable and suitable for the organization's reality.

1.2 Goals

The work presented in this dissertation follows two different, but interrelated approaches to enhance the security status of an organization by improving the SIEM capabilities. To achieve that, two objectives are set: 1) to establish a set of adequate Security Metrics to be applied within the SIEM system; 2) to develop a solution to gather information from public lists, classify its content trustworthiness considering external and internal information, monitor the organization's communications and reduce the number of false positives without decreasing the true positives cases rate.

The objectives are within two goals of the DISIEM's project: 1) providing OSINT information into the SIEM system, improving in the detection and prevention of new threats; 2) Defining a set of metrics every type of SOC and dedicated the SOC capabilities and Security Information management to monitor all the personal and infrastructure active

in the organization's security information.

The SM will increase the organization's awareness relative to their security status, and consequently, augment the knowledge about the risk within the organization. The OSINT solution will be capable of gathering information from public blacklists reporting suspicious IP addresses, assess the information with metrics to provide a reliable and a trustworthy output with a reputation score to be used with the SIEM rules. Enhancing the SIEM capabilities in monitoring the network communications of the organization.

1.3 Contributions

This work offers three main contributions: 1) a set of well structured and categorized security metrics; 2) new visualization prototypes to visualize the information from SM; 3) a new framework for gathering, assess and manage public blacklist using external and internal information.

As result of the project, EDP now includes OSINT in their monitoring process, from blacklists and security metrics to improve the efficiency of the SOC team. They started by creating their own public-private blacklist containing public and private information, including the results of alert investigation to assess the effectiveness of their sources. The incident response procedures were adapted to provide feedback so that each alert could be categorized as true positive and false positive. The introduction of OSINT, combined with continuous improvement of the Security Incident Management process, allowed the increase in the rate of malware detection while also reducing the number of false alerts, making the operations more effective.

The framework of the third contribution is divided into modules, and each module is independent from each other. The organization can choose the module more suitable for their status and priorities. 1) A program to gather suspicious IP from a set of blacklists. 2) An assessment program to analyse and classify over time the trustworthiness of the gathered IP addresses and blacklists. 3) A set of example rules to be used by a SIEM to reduce the false positives. 4) A management interface enabling the end-user to manage and monitor the blacklist, suspicious IP, cases related with suspicious communication and the organization's public IP from a graphical web interface.

As result of the contributions of this work, the article "Threat Intelligence: Usando informação sobre IP maliciosos para melhorar a eficácia de um sistema SIEM" was written and submitted to the Portuguese conference INForum [22] and accepted in the proceedings for publication.

1.4 Work Plan

The work began on the 30th of September 2016 and supposedly should have ended on the 30th of June 2017. However, due to the addition of objectives and the occurrence of unexpected issues, the work lasted until the month of August 2017. In this section the initial plan, the additional plan and the accomplished plan are described.

It is noteworthy that in mid-December there was a change of the initial plan with the addition of a new theme that would be the second component of this work: trustworthiness blacklists. This new module came with the need of the organization involved (EDP) to, at the time, searching for solutions to collect information from various public blacklists and using metrics to assess the content of these blacklists to reduce the number of false positives that the blacklist have a reputation for. The research and study on related work and the topic of SM had already been carried out, however this stage was extended to the study on blacklists, their collection, and their content evaluation and insertion into the SIEM systems to be monitored. With the combination of the two, i.e. Security Metrics and Trustworthiness blacklists, now the work was not only focused on SM for SIEM systems, but also on threat intelligence.

The work was completed in August with the writing of this dissertation, extending for another two months of its initial plan.

Figure 1.1 displays the vision of the course of this work with the initial (included the additional plan) and the accomplished plan.

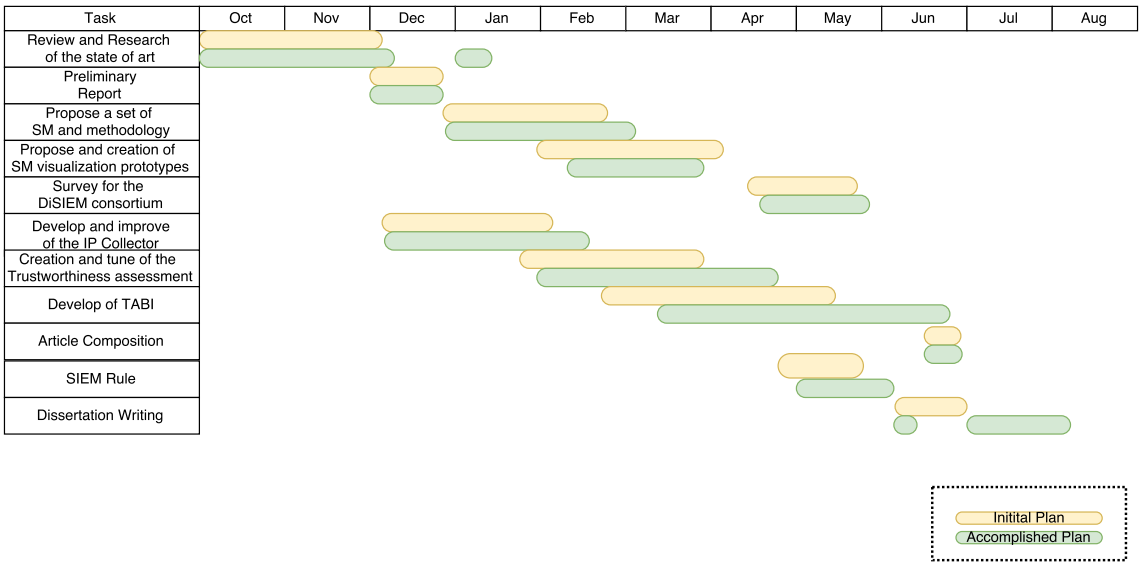


Figure 1.1: Work plan diagram

1.5 Structure of the document

The remainder of this document will be organized as follows. The next chapter introduces the context of this work. The chapter is divided in two topics EDP and SIEM system. The EDP's business and the SOC team are the focus of the first topic and for the SIEM's topic a detailed description about the SIEM used in EDP - HPE ArcSight - its architecture and features. Chapter 3 reviews the related work and discusses the current state of the art and a view about the areas of security metrics and trustworthiness of the blacklists. The challenges of the SM concept, how practical SM can be for the SOC and its implementation with SIEM systems. Chapter 4 is the developed work about Security Metrics for SIEM systems. This chapter presents the proposed metrics, taxonomies, principles and visualizations. These security metrics will help the information security team to manage their infrastructures and work flow, and the C-level managers to manage the SOC team and the organization's security resources. Chapter 5 - Trustworthy Blacklists focus in describing the developed framework of gathering public information from a set of public sources, the assessment of the information previously collected, and a web interface to manage and visualize results of the framework. In addition, the chapter provides guidelines to create rules to be used in a SIEM system to monitor and alert suspicious communications between the organizations and suspicious IP of maliciousness. Chapter 6 describes how the experiment was prepared, the environment it was submersed, and the analysis of the experiment's results. The document ends with a summary, conclusions of the results analysis and future research and developments to improve our work.

Chapter 2

Context

2.1 Energias de Portugal

EDP - Energias De Portugal, SA. is considered one major electricity operator in Europe. It is also one of Portugal's largest business group, the company was founded in 1976 after a fusion of 13 companies, and was the first Iberian company to own significant generating and distribution assets in both sides of the border. Currently EDP is the third Iberian major operator of renewable energies and one of the world's largest players in wind energy [13, 14].

Forbes Global 2000 magazine ranked EDP at position 437 in 2016 and is worth around 2.15 billion Euros, by a study conducted by consulting "Brand Finance", published in June 2016 [17].

Figure 2.1 presents the EDP's business and displays how complex the universe of EDP it is. EDP operates in three countries: Portugal, Spain and Brazil. In each country they have businesses in electricity production, electricity and gas distribution, commercialization and trading of electricity and gas.

2.1.1 EDP's Security Operations Center

EDP's SOC uses the typical components to enhance security and to reduce the risk. Firewalls, antivirus, and IPS are some of the components used. To link all the information provided by these cybersecurity appliances by monitoring security events, EDP uses the ArcSigh SIEM from Hewlett-Packard Enterprise (HPE) [21]. They also do awareness and countermeasure procedures to internal collaborators in the presence of cyber-threats. The SOC already uses SM to view the state of their tasks, to know the status of their systems and components and to monitor the number of incidents and vulnerabilities within the managed infrastructure. Periodic reports are produced with graphics about the security status of the company and applications to present to the C-level managers and the executive board.

EDP's SOC uses ArcSight's SIEM to monitor, manage (create, edit, delete) incidents,

Figure 2.1: EDP's business organization (EDP - Organização dos negócios, 2016)

create metrics, investigate possible security incidents, manage the devices, forensics and more.

The SOC's team thinks that the SIEM is not fully to its potential in SM and countermeasures, and there are some flaws on ArcSight SIEM. Although it has predefined security metrics and respective visualizations, the SIEM is limited concerning the creation of new visualizations making to improve the SIEMs plataform. These queries are interpreted as metrics, they will get measures using filters and other queries, then transform those measures into meaningful data for visualization. When a modification occurs and a query needs a simple modification, the work needed to perform, required a significant labour time, due to the imminent recreation of the query. All the dependencies from the queries associated with, and the visualization itself needs to be created from scratch. Another weakness in the ArcSight SIEM system, it is in the inflexibility of changing the close date of an event. When a member of the SOC team resolves an incident, and sets its date from open to close in the ArcSight, the value of the close date will be the current timestamp and cannot be modified. The value can be incorrect because the incident can officially be closed hours or even days before it was declared in the SIEM. These results are poor measures for metrics and reports. To bypass these two flaws of the ArcSight, EDP created an application external to the SIEM (internal in the network). The application is used to create graphics and uses other source besides the SIEM, providing

measures to be used by the SM, improving the accuracy of the results. Although the SOC team already thought about using OSINT technology to feed the SIEM, it was never fully implemented due the reputation of high false positive rate that the sources provide and the inflexibility of the SIEM.

Our solution aims to help EDP's SOC team to overcome these drawbacks, by developing and implementing new SM for the SIEM. These new SM will show information and status reports about the security and efficiency of the system, the reliability of the sources and support for the decision making. New visualization methods will be produced to present these SM. It will be developed a framework to gather suspicious IP addresses from public blacklists, by using the OSINT concept, assess the IP and the blacklists, using a correlation of the information collected and using the results of the company security incidents. The assessment will produce a more reliable output and with SIEM's rules we will monitor the communications between the assets of the company and the suspicious IP addresses. The conjunction of all the components of the solution will enhance the EDP's knowledge of their security status, increase the SOC's efficiency, and reduce the company security risk.

2.2 Security Information Event Management

A Security Information and Event Management (SIEM) system is a tool which combines the services of Security Information Management (SIM) and Security Event Management (SEM). Scott [19] states that the purpose of a SIEM is to gather and manage event log data. It collects and aggregates data to provide an effective and beneficial analysis capabilities for the information security team. With SIEM systems, the tasks of security managers - monitoring, incident response, reporting, investigating and auditing - will be more efficient, fast and accurate, due to the combination of SIM and SEM purposes.

The SIEM has six core functions: 1) Collects data from devices and from different types; 2) Normalizes all the data collected from the different vendors and devices to a common standard; 3) Enriches the event data gathered with taxonomies, network and assets with specific details; 4) Stores logs and events, and through a high compression ratio stores information of several years; 5) Searches all the information gathered with a simple interface and using a text tool; 6) Analyses all the gathered data in real time, identifies and traces data patterns to find threats and/or breaches.

2.3 ArcSight

The ArcSight is a HPE SIEM product and is used by the EDP SOC team. ArcSight SIEM encapsulates all features from a normal SIEM. ArcSight consists of three components that make its architecture: Connectors, Loggers and Enterprise Security Management

(ESM). In addition to these components it is also provided a graphical interface for the management. From the interface, it is possible to monitor, analyse and filter the data previously collected and processed. Creating custom filters and rules, having more than one active channel for each incident and automatic creation of charts are a few features that the ArcSight SIEM system offers.

2.3.1 Components

Each component of the Arcsight has a predefined task. The connectors collect, normalize and categorize all sources' information. The ESM and logger correlate and consolidate the information and display it to the user. The main difference is the storage capability, the logger has more space (providing a larger window's time of information), and the correlation engine. EDP decided six months of raw information for the Logger's storage and three months of filtered information for the ESM's storage. The flux of the SIEM's process, starting from the sources and ending in the logger and ESM user interface in represented in Fig. 2.2.

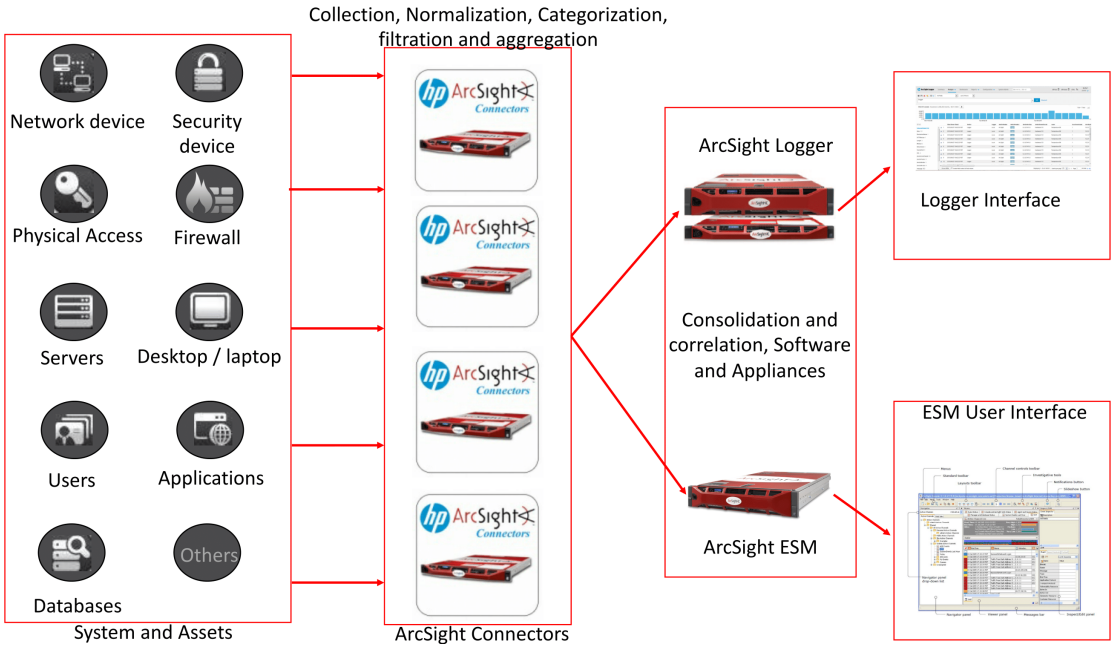


Figure 2.2: ArcSight Architecture

Connectors

A connector is an ArcSight software component whose purpose is to collect all the events, from a variety of sources, and forward to ArcSight destination components.

A connector is installed as an appliance or as a virtual machine, and collects the events from the logs of each connected device. The source device can be an IDS, Firewalls,

Databases, antivirus, operating system's logs, and more.

In the second phase, the connector normalizes all the distinct data to the Common Event Format (CEF). Each source device has its own standard log, therefore there is an extensive amount of information differently formatted and this step solves the differentiation between vendors.

The following phase is the filtering process, when the connector discards the unnecessary data previously collected. The filtering is needed because the connector cannot filter useful information on some devices when it is collecting (for example: Windows logs). Then all the similar information is aggregated into groups, providing a faster search, thus improving the performance. This last phase is called the aggregation phase.

To enrich each event with substantial information, ArcSight uses six criteria: the target object; the behaviour associated with event; the outcome - success, failure or attempt - the type of event (according with the security domain); the device group; and the sixth is the event's significance to separate normal events from hostile events. The connector concludes its process by sending all the essential information to the HP Arcsight Logger and the HP ArcSight ESM.

The Connector Appliance centralizes connector management and offers unified control of all the available connectors (a connector can be installed but not available, due to deactivation or malfunction). ArcSight Connector Appliance provides a single interface through which is possible to configure, monitor, tune, and update. This is desirable when the organization has a significant number of connectors. A connector's appliance can cluster operations and send all of them across to the connectors.

The SmartConnector is the default connector. It's an ArcSight software component, which collects events and logs from all its connected sources. In addition to normal connector features, the SmartConnector grants the possibility to add, remove and edit a smart connector, update the connector's table parameters, add and remove destinations, edit destinations parameter and send commands to a connector.

The ArcSight provides several types of SmartConnectors. Each of them contain a particular functionality. This dissertation only describes the FlexConnector, because of its ability to connect third-party devices.

FlexConnectors are custom Connectors that can read and parse information from the third-party devices and enrich the ArcSight's event standard. Some third-party devices do not have a log format known by a SmartConnector, hence it is necessary the use of FlexConnectors. Connector Appliance provides a development framework that lets the security team quickly and easily develop a FlexConnector, enabling test phases before deploying it. A security team member develops a FlexConnector by creating a parser file compatible with the target sources.

ArcSight Logger

The ArcSight Logger is a universal log management solution [20], which has an extreme high event throughput, efficiency in the long-term storage, and agile data analysis. ArcSight Logger collects logs and events of raw data from any logger generator source and storage a large quantity of logs in a simple management manner. Supports cybersecurity, IT operations and log analytic with quick searches and reports about the data or the investigated incidents.

The Logger also provides a web interface where its features can be used, and the security team can analyse and investigate the events. The Logger display those events in a tabular form, with fields that describe how the Logger received the respective event.

ArcSight Enterprise Security Management (ESM)

ArcSight Enterprise Security Management (ESM) is a software solution providing security event monitoring with network intelligence, correlation, anomaly detection, historical analysis tools, and automated remediation.

The ESM connects all the previous components for correlation of all the events collected and has flexible monitoring tools to investigate and remediate. It uses a workflow framework providing a structure of escalation level ensuring that events of interest will arrive to the security team members and in the right timeframe.

The ESM offers an automatic reporting tool, requiring a template document and indication of the fields to be filled with the ESM values. The template is uploaded to the SIEM and the security manager defines the creation date and the type for the report (monthly, quarterly, or another defined period).

The ESM uses other SIEM components and has its own sub-component to fulfil its task.

- SmartConnector: and their sub-classes (e.g. FlexConnector);
- Management centers: for a centralizing management of the connectors;
- Correlation Optimized Retention and Retrieval (CORR): Engine which performs high speed searches and process events with high rate;
- Data sources: all the sources connected to the connectors;
- ArcSight Manager: is considered the heart of the solution of ArcSight SIEM. performs analysis, correlation, workflow and services;
- User Interfaces:
  - ArcSight command center for all the manageable data, user, devices and services - not used frequently;

  - ArcSight Console to be used all the time by the SOC team for the daily tasks, using the ESM resources;

- Use Cases to view, configure, and transport developed sets of related resources which address a security issue;
- ArcSight Risk Insight is an add-on product that aims at providing information about the business impact of real-time threats to assets;
- Interactive Discovery is a separate software application that enhances the visualization (with dashboards, reports, and analytic graphics), data discovery and investigation of security data from the ArcSight platform.

Figure 2.3 is the ArcSight Console and displays some of the ESM resources. Apart from the multiple features in the top panel, this console can be divided in three main panels. The left panel is the navigator panel and it is where the active channels are displayed, organized by folders, and can be stored to be used in the future (the security manager can also choose to see the rules, case users, data monitors, as a drop-down list). In the right side is the inspect/edit panel where by selecting an event all the gathered information about that event is presented. The middle panel is divided into four sections, starting from the top and going down until the forth section. In the first section, there are six open active channels and presents a certain type of real-time events. The SIEM is filtering the events using the parameters given from the filters and/or rules to obtain the specific type of events. The ”Live” active channel is the channel currently selected. The second section is a summary about the active channel selected, this window presents the date and time (”start” and ”end”) that the active channel is getting the information, the used filters, the total number of matched events and the number of events divided by their severity. The third section is the radar active channel and shows the events and their severity over the defined time, where is possible to select a specific time frame. The last window is where the events are displayed, here the security manager can add tabs to know more information about the events. The console displays the ”Severity level”, ”End time”, ”Name” and ”Attackers Address”.

ArcSight ESM Resources

The SOC team uses the resources of the ArcSight Console as a support when analyzing, investigating and monitoring security events.

Although the ArcSight offers twenty-six useful resources (active channels, field sets, active lists, agents, assets, categories, locations, networks, vulnerabilities, zones, cases, customers, dashboards, patterns, reports, archives, rules, stages, users, data monitors, filters, knowledge base, notifications, partitions, patterns discovery and profiles) in this document the most relevant resources will be described.

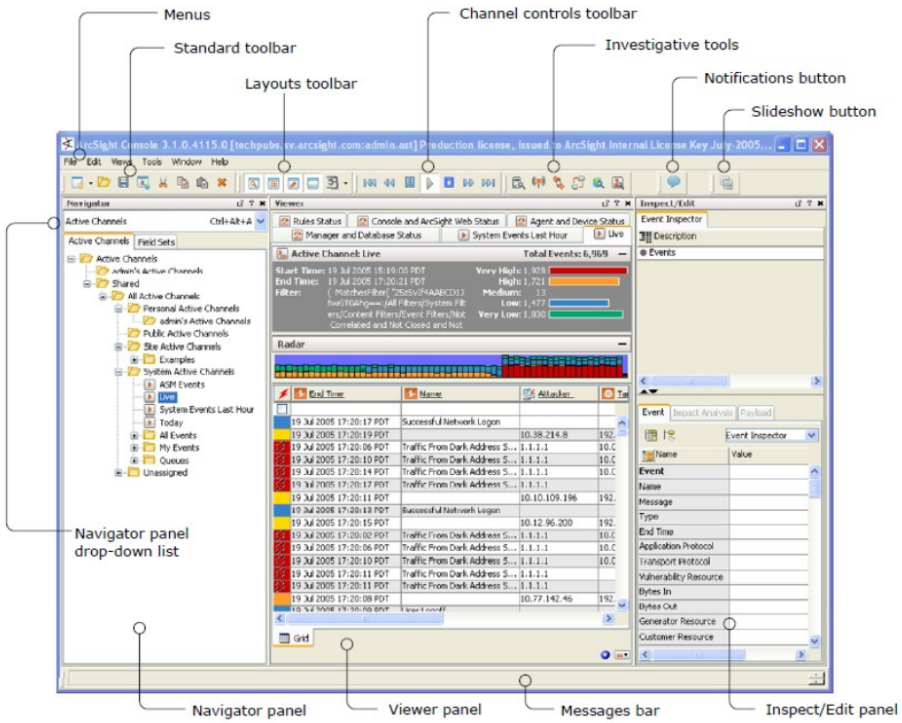
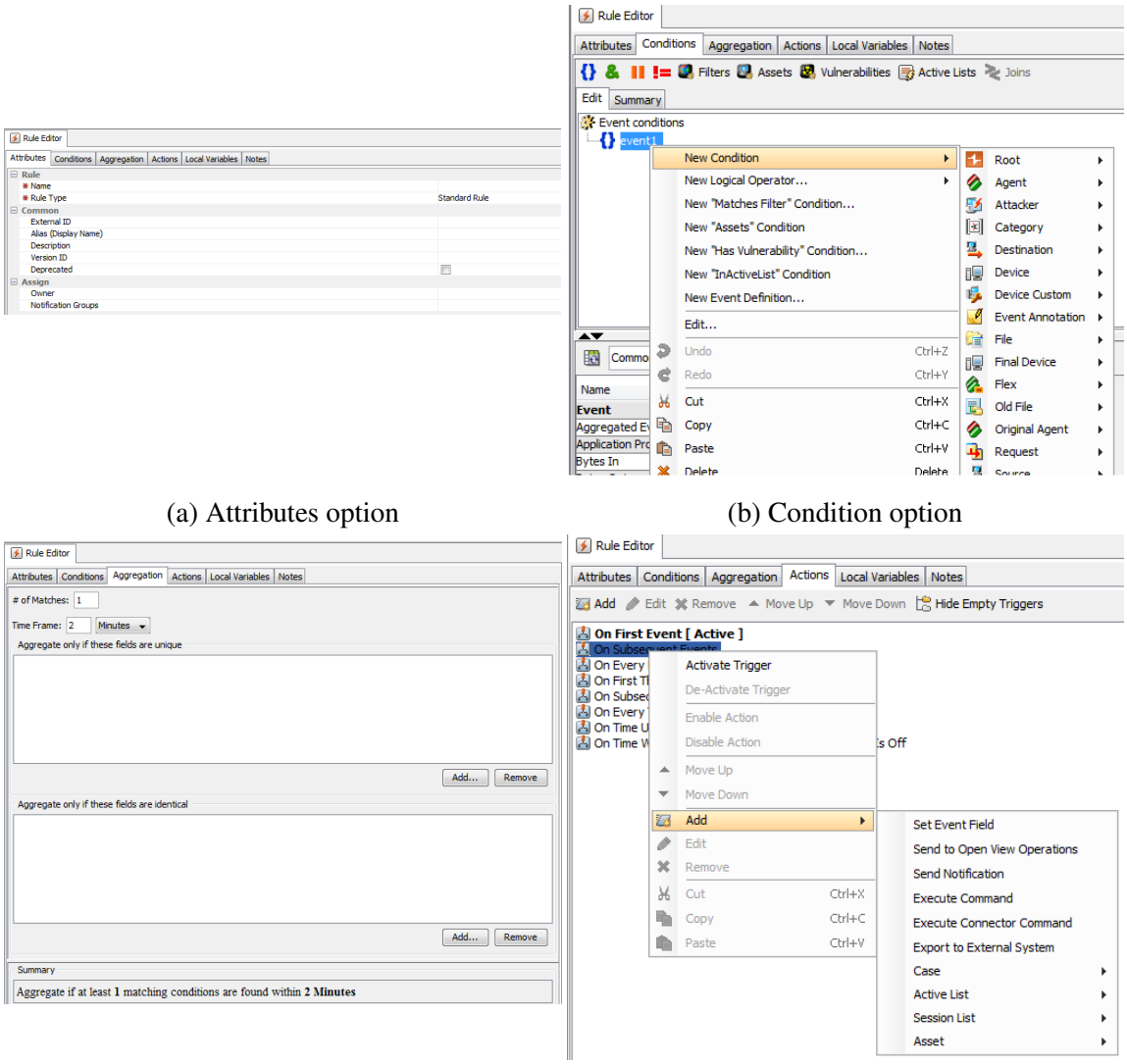


Figure 2.3: ArcSight Console View - retrieved from [11]

The active channels belong to the monitor view category and are real-time collection of events defined by parameters (filters, rules and date) created by the SOC analyst. Active channels contain two sub active channels: header and radar. The active channel header appears at the top of every single active channel and contains the statistical overview of the channel and the events passing through it. The active channel radar is a bar chart overview of events in the active channel. The events are sorted into segments by the event's end time. The grid view displays each event with a set of a data fields in a table view. The data fields are information about the events (severity, attack address, target address, etc.) and can be added or removed accordingly to what the security analyst wants to be displayed with the events. These three views are presented in the middle section of the Fig. 2.3.

Because we are going to use rules in the Trustworthy Blacklists component, we describe the concept of a SIEM rule in more detail that the other SIEM resources. A rule is a programmed procedure that evaluates incoming events for specific conditions and patterns, when there is a match it triggers actions in response. Helping the analysing and monitoring specific type of events. Figure 2.4 displays the available options when creating or editing a rule.

Figure 2.4a displays the basic options for a rule, such as the name for the rule, its description and the groups that will be notified by the rule. Figure 2.4b exhibits the conditions option, these conditions can be basic conditions, i.e condition also used by filters or active channels (these condition can be for example the *target address*, *target hostname*, *attacker hostname*, *attacker port*) or have the combination of filters, active lists



(a) Attributes option (b) Condition option

(c) Aggregation option (d) Action option

Figure 2.4: SIEM rule configuration options

(static or dynamic), association with only a set of assets and create a relation between the asset and the vulnerabilities known by the SIEM. Figure 2.4c is the third option is where it is set the aggregation events to trigger a rule. Here is where we define the events characteristics to trigger a rule. The options available are the number of matching events required to trigger a rule, the interval time to occur that match, the identical and distinct event's fields that are required to be considered a matching event. The final option is the Action option (Fig. 2.4d), this option set the action that the rule will preform when is triggered. The normal options can be a notification to the SOC, or a configuration over the SIEM, such as creating a new list from the result of the events.

The filters are a set of conditions that focus on an individual attribute of the event. With filters, SIEM reduces the number of events processed by the system. Filters also help analysing and monitoring some specific type of events in the correlation with rules

and data monitors.

When some conditions of Arcsight are triggered a notification is created. The notifications support the SOC team to monitor and to be alert on events, each notification contains the destination resource. The destination resource is the mechanism by which a security team member can add to an individual user or groups in the organization to receive a specific type of notification. The notification messages can be automatic and delivered by e-mail, text message, or by the ArcSight Console.

The Dashboards display indicators that communicate the state of the organization. Dashboards are made up of individual data monitors in a variety of graphical and tabular formats. To build a dashboard it is necessary to create queries. A query contains parameters, these parameters act like filters and select the essential information. The queries can have dependencies between them, is required to declare these dependencies and select them for the expected dashboard.

Figure 2.5 displays examples of dashboards with some predefined metrics. One of the dashboards is the 'Top categories', a bar chart type that shows categories of events and sort events by the number of times they match a rule. The "unknown" category contains the events for which the SIEM couldn't detect the name and categorization of the event. On the right side of the window, a pie chart displays the top target addresses. A query counts the number of times an address is considered as a target, and the results are presented in a pie chart. This pie chart only displays a visual distinction, by colours, about the different IP addresses. ArcSight can provide visualization of other metrics in addition to those illustrated in Fig. 2.5, such as the type of firewall rules triggered or the number of alerts by a rule.

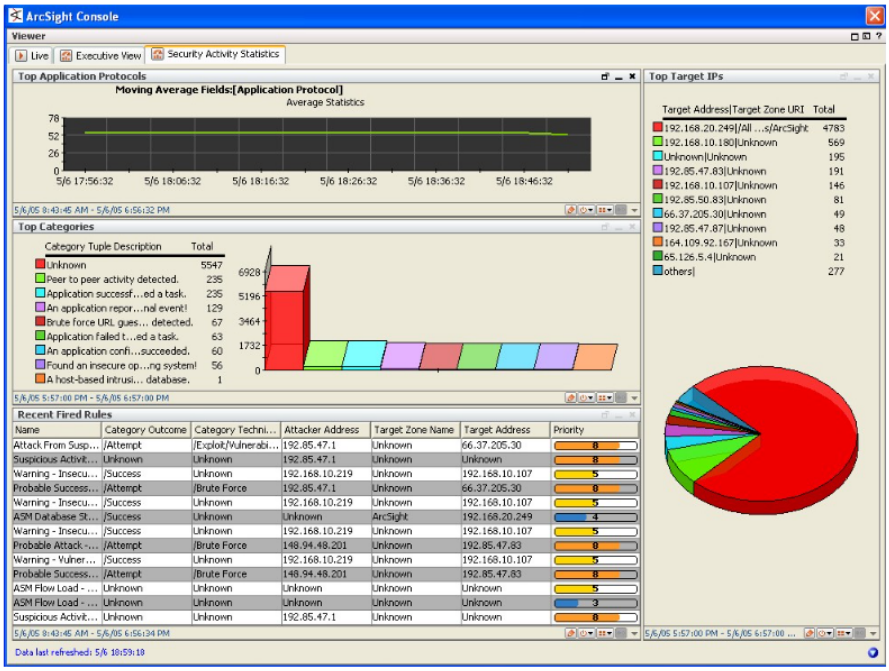


Figure 2.5: ArcSight Dashboards - retrieved from [11]

Chapter 3

Related Work

The use of Threat Intelligence in the organization is indispensable, nowadays, due to the valuable knowledge extraction that the information security team can obtain, and consequently, a better efficiency and time reduction in response to security incidents. Bromiley [5] defines TI as a fundamental process for the organization's information security defence. TI uses two factors for information discovery: external and internal. The external is characterized by the discovery of threats outside the organization, provided by feeds (social networks, blogs, forums, security communities or paid subscriptions), information sharing by government, police forces and security organizations or organizations of the same sector or geographically closer. As for internal discovery, the goal is to gain detailed knowledge about the level of security of the organization. Detecting system vulnerabilities, monitoring and detecting security anomalies, and deviations from normal behaviour are aspects which help to know the organization's security status. The following sections review works related with these two topics, their theories, developments, results and conclusions. They will be the foundation and initial principals for the developed work.

3.1 Security Metrics

How can we defend ourselves if we do not know our own weaknesses? Notwithstanding the importance of knowing the outside threats, all that awareness is insufficient if we do not know about our security status. Security metrics are the solution to do an efficient, precise and objective internal discovery. All the types of organizations (academic, government, and companies) are studying the SM to provide more precise and complete information about the systems security and risk status of the organization.

3.1.1 Definition and Purpose

One of the problems related to the development Security Metrics is the ability to incorrectly define SM. A bad metric definition leads to misinterpretations, which originate inappropriate evaluation and by consequence a wrong risk assessment. Therefore, instead

of an improvement of security and a risk reduction, the opposite is obtained. However, defining SM alone may not help in deciding whether the metric that security team chooses is a SM, and whether it responds to the purpose of the organization. Using both (definition and purpose) is the most proper option to obtain the desirable results.

Jansen [23] and Jaquith [24] state that the definition of Security Metrics is the measurement based on quantifiable measures and is a manner to put numbers around activities of security information. SM are a subsection of metrics and specify which quantifiable measures must be security-related, maintaining linearity and the methods well defined. Payne in [33] goes further and separate measurements from metrics, saying that measurements are raw data collected and metrics are either objective or subjective human interpretation over measurements, but always simple and precise. Metrics can be an efficient tool for security managers to notice the effectiveness of their security programs and their components. With the knowledge gathered through metrics, security managers can answer questions such as, "are we more security today that we were before?" or "are we secure enough?" or even "how secure are we?". Others author link Security Metrics with measuring risk levels and countermeasure decision-making. Julisch [26], and Kaur an Jones [27] define SM as valid and precise functions, whose return values are inversely related to the vulnerability of the measured system. SM are tools to identify the adequacy of controls, to provide a baseline for comparison purposes, to evaluate the security built, and provide financial information. This management makes better information security decisions. In the same work Julish, also proves consistence of this definition in the field of software quality metrics.

Jansen, Jaquith and Payne have the same concept of SM's purpose, described in [23], [24] and [33], respectively. Them, Muthukrishnan and Palaniappan [31], Rathbun [34], Tashi and Ghernaouti-Helie [39], argue that Security Metrics as vital role to any organization. The SM's purpose is to provide an understand about the security risks, to discover potential problems in the system, detect failures in the IT controls, weakness of the security infrastructure, measure the performance of countermeasure and process, facilitating the decision-making. In addition, SM strive to offer a quantitative and objective basis for security assurance for strategic support, quality assurance, and tactical oversight, also provides more information for the assets' accountability, These criteria can be achieved with models and algorithms which are applied to a collection of measured data.

3.1.2 Gathering and Generating Metrics

As will be explained, there are two methods to generate metrics. It will be used the top-down approach to explain the related work of gathering and generating metric.

Raw Data

There are many devices that can provide useful raw data to the SM, yet where, how and what can be challenging questions. These same devices can also give wrong information so is necessary to take care and exactly know how to answer these three "simple" questions. The correct answers will help to discard the unnecessary and unusable raw data for SM and if it is not possible to gather the right information discard too complex and not feasible SM.

The works of Berinato [4] and Vaarandi and Pihelgas [41] answer these questions. The use of network scans to find devices and have better understanding of the network's structure provides network coverage. As Berinato states in [4], the network discovery is an optimal tool to use and the raw data will provide good security metric. To extract valuable information from the logs, the work in [41] is explained the necessity to filter and remove the duplicates, reducing the large amounts of duplicate and unnecessary raw data collected. The normalization is necessary when the organization has different types of logs. The correlation between the logs will also provide credible and more complete data for the SM. The authors work goes further and each process is to explain how beneficial are the logs and understanding what information each log provides, knowing what raw data the SM needs and which logs provide that data, it some fundamental criteria and helps in the selection of desirable devices. Such as the logs of IDS and detect their false positive if an important data to determine the flaws or wrong configuration in the IDS. Therefore, SM can also be helpful for collecting correct raw data. The SIEM already collects, normalizes and correlates the logs. However, is necessary to define which logs should be feeding the SIEM, to not have a considerable volume of unnecessary information.

Good vs Bad Metric

Having valuable raw data is important, yet if the generation and selection of metrics is not done with care, all the raw data collected will be useless and meaningless SM. First is to know how to differentiate a good metrics from a bad metric. Jaquith [24] describes a list of good metrics and bad metrics, so the security manager can check which side his metrics belongs to. Good metrics should satisfy five criteria: 1) Consistently measured, without subjective criteria; 2) Cheap to gather, preferably in an automated way; 3) Expressed as a cardinal number or percentage, not in a qualitative label like "high", "medium" and "low"; 4) Expressed using at least one unit of measure, such as "defects", "hours", or "dollars"; 5) Contextually specific, and relevant enough to decision-makers that they can take action. As for bad metrics, in the same work, Jaquith considers those that are inconsistently measured, usually because they rely on subjective judgements that vary from person to person, cannot be gathered cheaply, as is typical of labour-intensive surveys and one-

off spreadsheets, and do not express results with cardinal numbers and units of measure, instead, they rely on qualitative high/medium/low ratings, traffic lights, and letter grades.

Payne [33] uses an acronym for security managers to know if a metrics is good to use: "Good metrics are those that are SMART, i.e. Specific, Measurable, Attainable, Repeatable, and Time-dependent". SM that are SMART indicate the degree to which the system is from the security goals. Building a SM program can be difficult and sometimes is possible to deviate from the objective.

Proprieties to select SM

To select and generate metrics is necessary to establish some proprieties. Otherwise, will be created metrics which are not the focus of the work and don't give valuable information. Rathbun [34] describes that all SM that answer a question nobody is asking are to be discarded. Here again is necessary to understand the organization's security objectives. SM provide decision support and nothing else.

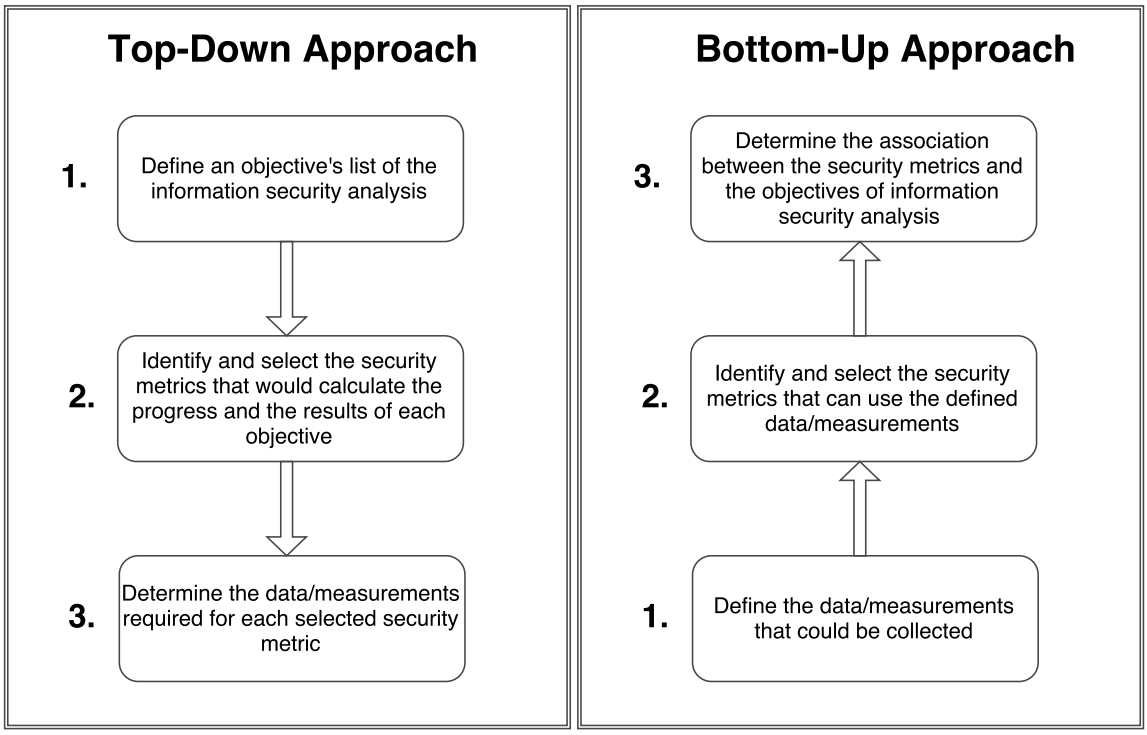
In the works of Rathbun [34] and Tashi and Ghernaouti-Helie [39] is explained that gathering SM can be efficient and easier if some simple questions are correctly answered: what data is to gather? Why gathering this kind of data? How to collect the data (programs, logs, etc.)? When to collect the data (frequency)? And where to collect (which devices/assets to tap)? The answers to these questions will give a good foundation (whoever not a technical or direct) guide about metrics. This "guide" can be also used for SM. The security manager needs to know the organization's security objectives and which departments will the collected metrics be presented to. If he wants to demonstrate a financial aspect for the board and executive, financial metrics are requested, if is for the operation team, technical measurements are needed. If the responsive team doesn't know how to measure, the probability of the final results with wrong values will be high. Therefore, the main goal is to obtain reliable and understandable measurements, selecting only what is really important and is according with the organizations' security objectives. Vaughn et al. [42] also agrees with this. They state that governmental metrics should be addressed for upward reporting and organizational report. As for the commercial side, their metrics are more focused to answer questions about how strong is the security perimeter, what is the return of the investment (ROI), etc. Last, but not least, Jansen [23] specifies five matters that should be in mind when selecting metrics, which are: Correctness and Effectiveness, Leading Versus Lagging Indicators, Organizational Security Objectives, Qualitative and Quantitative Properties, Measurements of the Large Versus the Small. All these proprieties should be previously chosen, if it is done the selecting phase will be easier and efficient. Besides, gathering and knowing what is a good and a bad metric, to select SM is necessary to establish some proprieties.

Approach to generate Security Metrics

To create SM is necessary to follow some guidelines. Two approaches can be adopted from [33]. Even not knowing many organizations already use one of these two. Figure 3.1 displays the two approaches described in the article. The first one (Fig. 3.1a) is the top-down approach, and starts by the information security team defining the objectives and then goes to select the necessary metrics that would help reach these objectives and finally find the measurements needed to generate those metrics. The big advantage of this approach is that identifying the metrics that matter will take less time.

The second is the bottom-up approach, as illustrated in Fig. 3.1b, is the opposite. The security team starts identifying the sources for the measurement, following by generating the metrics that are possible with the collected measurement. And lastly, evaluates if those metrics would help to the final goals. The advantage for bottom-up approach is the easiest way to obtain the metrics.

Both approaches are recommended to use when no framework is implemented.



(a) Top-Down Approach for the SM (b) Bottom-Up Approach for the SM

Figure 3.1: Two approaches to generate security metrics

Following or knowing about these steps is essential for the security of information, yet they aren't strict and can be modified accordingly to be more convenient.

3.1.3 Categorization, Classification and Taxonomies

Taxonomy is a classification scheme and helps in the classification and management of the organization's SM. With a well-defined taxonomy, the metrics that have been created will be more efficient and useful to the organization. If a SM do not fall under the classification should be discarded for the simple reason that they are not necessary or will not be useful. If the team thinks that one metric doesn't fit under the classification of the taxonomy but is important, then the taxonomy should be revised. Taxonomies improve the cooperation within the teams, even if they belong to different departments.

The classification of metrics may vary among organizations, even if they use the same methodology. Jaquith [24] states that we can use standards as a guide to build frameworks, yet the organizations shouldn't misuse taxonomies and must create to the organizations structure.

The work in [9] provides twenty metrics definitions specific for business functions. Business functions for (Security) Metrics area is a set of functions in each contains a set of metrics to help fulfill the functions purpose. [9] provides seven business functions and respective metrics. Table 3.1 (based on the information available by [9] presents the each business functions and purposes, respectively.

Function	Purpose
Incident Management	Determines how well the organization detect, identify, handle and recover from security incidents
Vulnerability Management	Determines how well the organization manage its security exposure by identifying and mitigating known vulnerabilities
Patch Management	Determines how well the organization are able to maintain the patches state of its systems
Configuration Management	Presents the configuration state of the system of the organization
Change Management	Determines how the changes of the system configuration can affect the security of the organization
Application Security	Determines the reliability on the security model of business applications to operate as the organization intended
Financial Metrics	Evaluates the investment made in information security

Table 3.1: Business functions and their purpose - derived from [9]

The [9] also categorizes Security Metrics in three hierarchies, based on their purpose and audience. Table 3.2 presents the categories with the functionality and audience.

Metric Category	Functionality	Audience
Management Metrics	Provide information about the performance business functions and the impact on the organization	Business Management
Operational Metrics	Improve the tasks of business functions and a better understanding	Security Management
Technical Metrics	Provide technical details and can be a support for the other metrics	Security Operation

Table 3.2: Metrics Categorization - derived from [9]

IBM also created his own taxonomy, as shown in Fig. 3.2 - [26]. The purpose was to create a new classification type of security metrics. This classification – unlike the previous ones – is based on the input data analysed by the SM. The decision to use input data as the basis of a new classification was made because has a particularly large influence on validation, accuracy, and precision for SM.

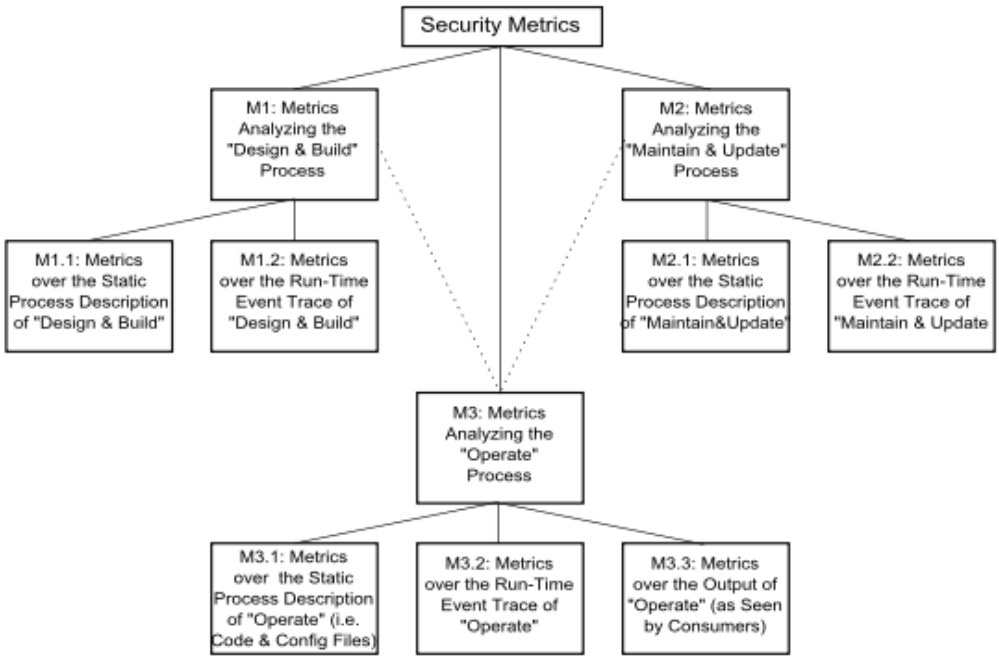


Figure 3.2: IBM Taxonomy: Classification of Security Metrics by their Input Types - retrieved from [26]

Based on the evaluation of some proposed taxonomies, Savola [36] proposes a high-level information security metrics taxonomy which covers metrics for organization information security and product development.

Figure 3.3 and Fig. 3.4 display two examples of the proposed taxonomies. Figure 3.3 illustrates a taxonomy for business-level SM with two levels (0 and 1) and Fig.3.4 shows a more detailed taxonomy for SM for information security management with three levels. The number of levels depends on the detailed level the organization wants to work with.

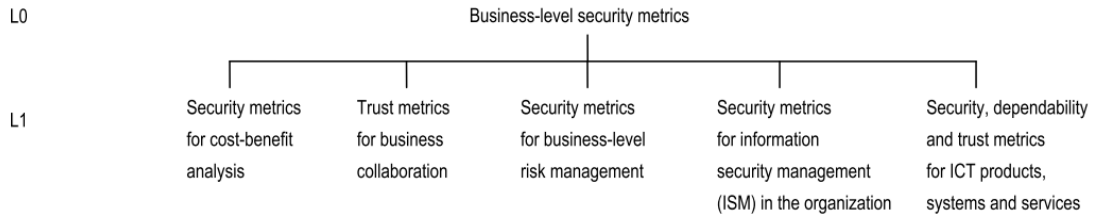


Figure 3.3: Business-level Security metrics (levels 0 and 1 taxonomy) - retrieved from [36]

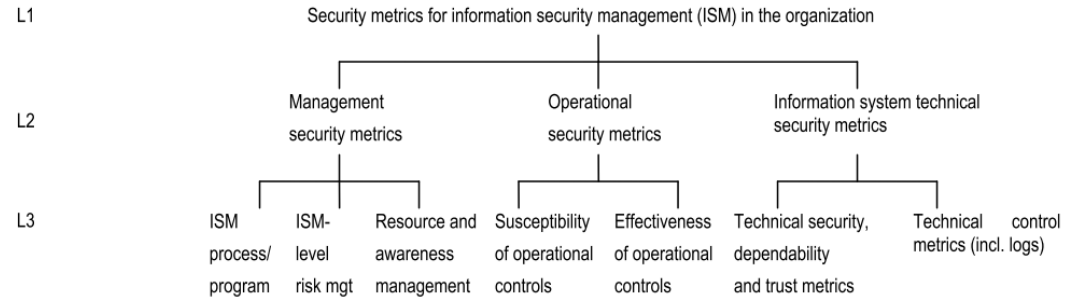


Figure 3.4: Security metrics for information security management in the organization - retrieved from [36]

3.1.4 Visualization

SM can provide useful information, yet if we can not interpreted and show its results, the SM can be misinterpreted or too confuse to understand. Explaining and showing the results of the SM to the C-level managers can be an handicap. The different levels of technical language domain and the vast quantity of information to present, can be a obstacle and turn into confusion and misleading the interpretation of the results. Jaquith [24] refers these problems and, like Kotenko and Novikova, Payne, and Rathbun, in their works [28], [33] and [34], respectively, suggests a way to transform the hard work-data to an elegant and clean way to present to the board.

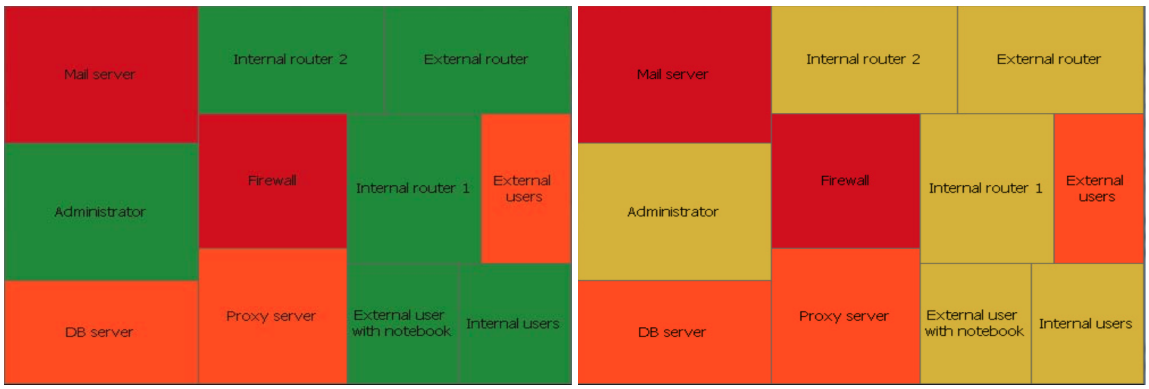
Jaquith [24] recognizes that in the ”practical world” the board members prefer qualitative, ”traffic lights” pie charts methods. But he refutes these methods, due to their tendency to be graphically inefficient and oversimplify issues too much.

He also mentioned some charts/graphics that are a good choice to select, for example: waterfall charts, time series charts, two-by-two matrix, etc. Sometimes qualitative metrics are more important than quantitative metrics. But if qualitative metrics were originated from quantitative metrics, based on the rules of [24] these metrics are not "good" metrics but can be used to represent the quantitative metrics in a more pleasant view.

Kotenko and Novikova [28] propose a visualization technique to represent a set of security metrics. The SM were used to measure the network security status and evaluate the efficiency of the protection mechanisms. While creating this technique, the goal was to assist and solve security tasks (given by the SM) which are important to SIEM systems. There are two visual model designs presented in this work that are worth to be mentioned: treemaps and security metrics graphical representation.

Treemaps

Treemaps is a technique used to analyse the possible consequences of attacks and countermeasures. Using interactive treemaps is possible to represent both a vulnerability report and a network security report. Figure 3.5a and Fig. 3.5b display two examples of treemaps. In these examples the treemaps were used to analyze the network security level. The business values of the host (asset) define the rectangle size, and the corresponding colour is the result of calculating the host security level or severity of the vulnerability. With treemaps the security team can immediately gather the most important problems as these maps also help to identify the risk of each sector in the organization.



(a) Security metrics for information security management in the organization

(b) Security metrics for information security management in the organization

Figure 3.5: Examples of the technique treemap - retrieved from [28]

Circle-based Pictogram

The circle-based Pictogram is a combination of two images to compensate the problem in which the user had to switch between the two treemaps (or other types of graphics) to compare them. The circle-based pictogram enables the division of **N** sectors and, thus,

provides values of **N** metrics. The outside ring represents the previous values of the metric (hence is more simple and fast to compare). Figure 3.6 shows a host representation using this technique.

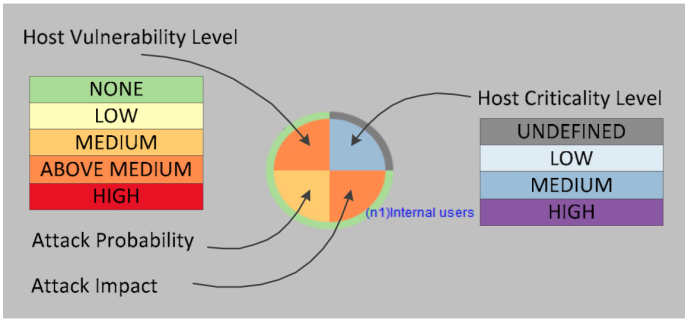


Figure 3.6: Security metrics for information security management in the organization - retrieved from [28]

Although the authors considered this technique to implement countermeasures, the technique can be used in a whole different level. For example, using the developed chart to show the number of vulnerabilities open, closed, open this month, closed this month, and compare these numbers with the number from the previous month, for visual standards, is easier than two circle charts. The security team can add outer rings. Each outsider ring represents a previous month. Implementing this technique in the SIEM dashboard could be complex or even impossible to accomplish, due to the inflexibility of the SIEM. However is possible to implement this feature into the EDP's external application.

3.1.5 Metrics used for threat intelligence

One mechanism of threat intelligence is the internal discovery. The appropriate approach to an organized, accurate, and objective discovery of internal information is the use of security metrics. Currently there are several articles that provide a set of metrics that can be used within a SIEM system. In [3] presents a set of security metrics to be applied with (or even on, using certain features) SIEM, namely ArcSight. In the same article is presented the metric: Quiet Feeds, which describes the correct functioning of the sources and that counts the number of sources that are not sending data. This metric can be used to discover internal information to know which sources (logs, antivirus, IPS) attached to the connectors are not sending information. It can also be used to identify which blacklists may not be providing information relevant to the organization.

A work that uses outside knowledge (without OSINT) and knowledge of the organization is the work of Kotenko et al. [29], which creates models about the impact of a threat on an organization. The authors design a model that identifies which asset (or set of assets) of the company is subjected to a threat by combining the use of security metrics to identify vulnerabilities in assets and dependencies between them, and the knowledge of

a threat. In the model it is possible to know the risk value for each threat and the surface attack which presents how wide an attack can be for an organization. However, this is not done autonomously, requiring a human investigation into new threats and collecting information on the organization's vulnerabilities.

3.2 Trustworthy Blacklists

The organization enhances its security by knowing its own weakness and flaws, however is also necessary to know the external threats in order to obtain intelligence about the existent external risks and, with that, prioritize and implement the required cyber defence measures.

3.2.1 Open Source Intelligence

The open-source intelligence (OSINT) is the typical concept used for external discovery. Johnson [25] describes the importance of OSINT and the capabilities of the technologies that use it. These technologies allow the information security team to intelligently capture and correlate information from the Internet, producing a valuable result for the organization. Security feeds are a good source to obtain information about external cyberthreats and with the use of OSINT it is possible to collect pertinent information from several public feeds [5]. A blacklist is an example of a public list which contains information about cyberthreats and malicious behaviours.

One program that can collect multiple blacklists and has the possibility of a specialized configuration by the organization is IntelMQ [16]. IntelMQ's main feature is to collect and process security feeds, such as logs, tweets, or blacklists, in an autonomous manner. This tool enables the information security team to efficiently collect information from a set of feeds. However, if the information we intend to collect is different from the standard syntax, it is necessary to create modules, or use similar modules, to correctly collect information from each intended source. After all the configurations and all the requirements are completed to collect the information from the public lists. Is necessary to gauge to which extent feeds are trustworthy and if indeed it is possible to rely on them, based on the information obtained to implement defence mechanisms [5], and the IntelMQ does not have that functionality.

3.2.2 The efficacy and trustworthinesses of Blacklists

There are some articles that investigate the effectiveness of blacklists and which, in a period, provide the most reliable information. Blacklists contain a significant rate of false positives [30, 35, 37]. However, it is known that information acquired from a blacklist is a measure widely-used for monitoring and detecting malicious behaviours [30, 37].

Sinha et al. [37] analysed four blacklists (NJABL, SORBS, SpamCop and SpamHaus), which report suspicious email addresses considered as spam. It was used an unsolicited mail detection program for the confirmation and detection of false and true positives. After analysing email traffic in an academic environment (more than 7,000 computers) within 10 days, the results confirmed that blacklists contain a significant number of false positives.

Kuhrer et al. [30] aim to understand the content of the blacklists and how its information is collected. They present two mechanisms: the detection of parked domains and the detection of sinkholes. They propose a mechanism to distinguish parked domains from benign domains, thus reducing a considerable number of non-benign domains present in a blacklist. It is also described a method for the detection of sinkholes, using a technique developed by the authors (graph-based), and their removal in blacklists. Sinkholes are, for example, servers that contain malicious domains, but have been controlled and mitigated by security organizations, which use them to monitor the network and communications with malicious domains. The authors conclude that blacklists only contain about 20% of malicious domains, resulting in a significant number of false positives.

In both previous works, it is complicated to state correctly and over time whether the effectiveness of a blacklist will increase or decrease.

3.2.3 Blacklists without trustworthiness

AlienVault's OTX [2] is a tool similar to the one developed in our work. It gathers information about IP addresses through reports by a set of communities. After a collection, the threat of the denounced addresses, is assessed considering the number of attacks, the number of denunciations and the type of maliciousness to which the suspected IP address is associated. The result is a list of IPs that can be used for monitoring or blocking IP addresses with a threat value calculated by OTX. However, the assessment it is only made for the IPs that are in the OTX and not for the blacklists chosen by the organization's security team. On the other hand, it does not consider the organization's cases to revaluate the reputation value of each IP.

3.3 Summary of the chapter

This chapter described works in the fields of SM, OSINT and public blacklist. In the field of SM exists an ample research over metrics, such as guidelines to create, maintain and discard meaningless metrics, there were works about metrics which can be used with a SIEM system. The field of OSINT with blacklists is a theme which got more studies over the last years, due to the security information which the blacklist provide. However, the investigations were only about the gathering or only about the manual assessment of the blacklists credibility, without considering the organizations' reality. After the study

of the related work we can establish a set of well-structured SM to be applied within the SIEM system, with a taxonomy and visualisation examples. These proposed SM will be evaluated by SOC teams from different organizations and use different SIEM systems, to verify and select the metrics that can be used by all the SOC and can be implemented in a SIEM.

Chapter 4

Security Metrics for SIEM systems

"If you know neither the enemy nor yourself, you will succumb in every battle", [40]

If we do not know about ourselves, our weaknesses and our strengths, we become an easy target of malicious intents. How can an organization know about itself in terms of security, especially worldwide organizations such as EDP? Security Metrics are a suited tool for knowledge enrichment about the organization's security and can answer questions about the organization's strengths, weaknesses and risks, with an overall of the organization security status.

This chapter describes the developed work in gathering inside information about the risk and security status of the organization accordingly with the different SOC capabilities and the different SIEM systems. We propose a well-structured Security Metrics with a precise definition and purpose, a taxonomy for the SOC capabilities and prototypes to enhance the visualisation of SM.

4.1 Definition

To understand SM it is essential to have a proper definition about them and they must be within the goals of the information security team. For the security information team, security metrics are the final step of measurement and provide information about the system security status (and other related information), providing substantiated information to the cybersecurity manager to have a wise decision-making process, resulting in the enhancement of the system security. The security enhancement can be achieved by changing the definitions or policies, countermeasures or resources reallocations.

4.2 Taxonomy and Methodology

Security Metrics should be organized to not deviate from their purpose, thus will be easier to discard the SM which are unsuitable for the organization. A methodology must be well

defined in order not to repeat steps and to not take unnecessary working hours in the SM's creation and maintenance.

The Taxonomy we created is similar to [1]. We divided the SOC capabilities into three main categories: Management, Process and Technologies - as illustrated in Fig. 4.1. This taxonomy follows the normal standard of an organization hierarchical structure and SOC capabilities.

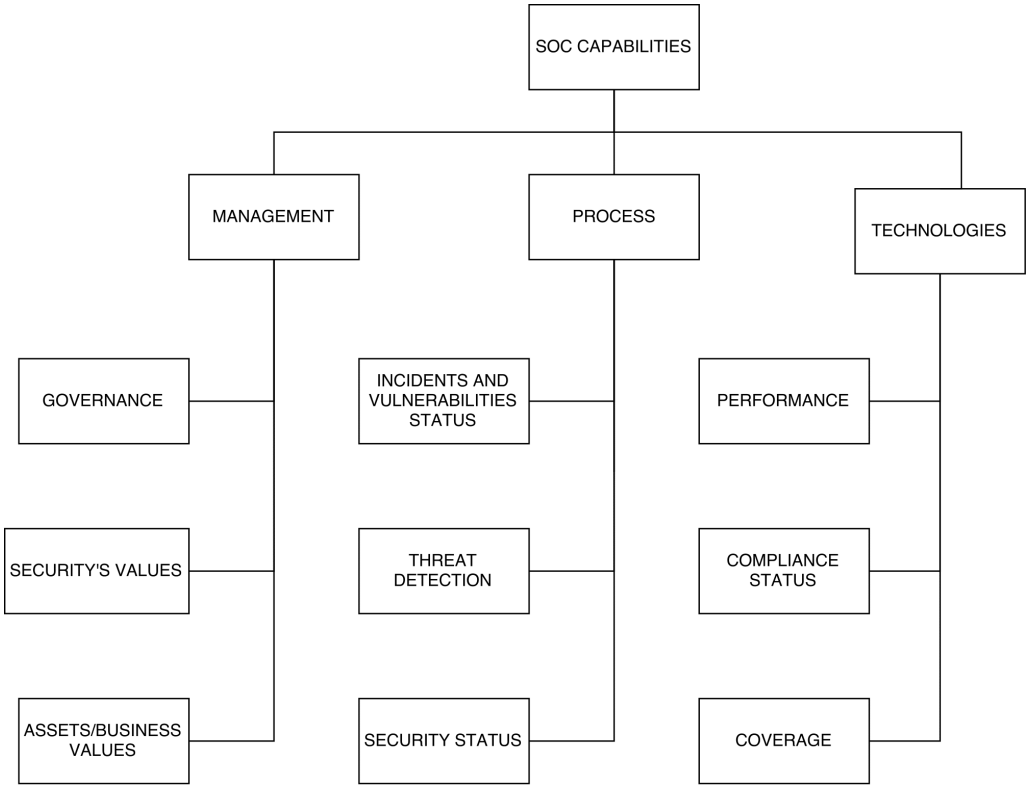


Figure 4.1: Taxonomy for the SM following the Capabilities of the SOC

The target audience for the management category are the C-level managers. The metrics for the Management category are inserted in three subcategories. Governance, providing information about the administration and management over the workers and external providers within the organization network. Security's value, contains metrics about the investment made and the return of enhancing the security. Organization values, related with the assets and business value of the organization and the cost of their loss.

The Process category focus is to provide a more manageable control over the incidents and vulnerabilities, the communications and the state of security in the organization. The Process category is divided in three subcategories: 1) Incidents and Vulnerabilities status, where the metrics about incident and vulnerabilities detection and resolution; 2) Threats detection, metrics about detection of anomalies and abnormal behaviour; 3) Security status about the system and subsystems.

The final category is Technologies, which focuses on the correct and incorrect op-

eration of the cybersecurity tools. The Technology category contains a set of metrics to calculate the Performance, Coverage and Compliance Status of security tools used to monitor the assets, detect anomalies or malicious behaviours.

The subcategories are used as topics to confirm if a metric corresponds to one main category. With them it is possible to determine if the new metric should be discarded in the organization or the taxonomy should be reviewed.

In this work, the Top-Down Approach was used, and the main SM objectives were established: 1) An overall of the organization's risk; 2) individual risk over the system and sub-systems; 3) information about the cybersecurity appliances security and operation status; 4) information about the SOC team effectiveness; and 5) advise on whether, or which, investment (monetary, staff, work labour) is worthy considering the organization's status and possible loss scenarios. The next section will describe the two next steps of the Top-Down Approach, the selected SM and the sources for the data/measurements.

4.3 Proposed SM

We did an investigation about which SM are most used and appropriate for SOC teams and SIEM systems.

Upon completion of the investigation, we surveyed a total of 63 security metrics that we think they are appropriate and dedicated for SOC teams and SIEM systems. 10 SM are for Management, 32 for Process, and 21 for Technologies. From those 63, 15 were variations from the originals and 2 are new metrics with 6 variations, the remaining were retrieved from [3, 4, 6, 7, 8, 10, 19, 29].

Appendix A presents all the 63 SM. This survey, with a questionnaire, will be for the consortium of the DiSIEM project and the answers will be analysed to verify which of the selected SM can be implemented and which sources can produce the required data for the SM, and if the SIEM (or third-parties) can provide the information from the sources. The survey will be published as part of the deliverable 3.1 of the DiSIEM project. In this section it will be just explained the three metrics created, their variations and the metrics that it will be used in other modules.

The metrics created are the PETVI (Sect. 4.3.1) and the ERVIDENT (Sect. 4.3.2). These two are the inputs for the TPerf (Sect. 4.3.3). Their results, included the TPerf, should be kept in history to be compared for the following months, thus allowing observing and correlating of the teams' effort and efficacy in resolving vulnerabilities and incidents.

4.3.1 PETVI

PETVI - SOC's Percentage of effort Time to resolve Vulnerabilities and Incidents: it calculates in percentage, the SOC's team effort time to resolve the vulnerabilities and

incidents which were opened in that period. The input data for this metric is the SOC's team total work time and their deliverable time to resolve the vulnerabilities. The output will be a percentage of the team's effort time to resolve the vulnerabilities and resolve incidents.

$$PETVI = \frac{t_{vuln} + t_{inc}}{t_{total}} \tag{4.1}$$

Where,

- t_{vuln}: The number of the team's hours to resolve vulnerabilities
- t_{inc}: The number of the team's hours to resolve incidents
- t_{total}: The total number of hours of the team

Due to the SOC's operations, we created three variations of PETVI. Two of them are for only calculating the effort to resolve vulnerabilities or incidents. The input data is amount of work labour time if of the SOC's team and the amount of time spent resolving vulnerabilities or incidents, respectively. The third variation covers the scenario which the metric PTVI does not includes. We observe that, at EDP, the SOC's operation sometimes resolves vulnerabilities and incidents which were not opened in the PETVI evaluated period, so the metric PETV does not consider the team's time spent to resolve these cases. This would bring discrepancies to the results. The third variation, in addition of considering the same scenario has the PETVI SM, also considers the vulnerabilities and incidents which were resolved in the evaluated period, but were opened in a previous month.

The C-level managers can use the results of the PETVI and its third variation to do a well-conducted assessment over the effort time of the SOC's team.

The PETVI and its variations should be calculated monthly. They are considered in the Management Category, in the Governance subcategory, because they manage the effort time of the SOC's team.

4.3.2 ERVIDENT

The metric ERVIDENT, Efficacy of resolution of vulnerabilities and incidents, calculates the efficacy, in that month, of the SOC team and other teams involved in resolving incidents and vulnerabilities which were opened and closed in that month, and calculates the efficacy of the SOC team and other teams involved in their resolution. The input data are all the cases of that period and the output will be the ratio between the total cases resolved and the total opened cases of that period. The metric computation is given by (4.2).

$$E_f = \frac{R_C}{Total_C} \tag{4.2}$$

Where,

- E_f: Efficacy of the team resolving cases
- R_C: Resolved cases in that period
- $Total_C$: Total cases in that period (resolved cases on that month + open cases that month)

For the same reasons as the PETVI metric, the Efficacy of resolution also has three variations. Two variations only consider the vulnerabilities or incidents. The third considers the vulnerabilities and incidents which were closed in the evaluated period, however were opened in a previous month.

The calculation frequency of the metric and its variations should be monthly. This metric, and its variations, are included in the Process category, subcategory Incidents and Vulnerability Status. It could be used in the Management category but the objective here is to verify if the work flow of the SOC and the security tools used - the whole process involved - in detection and resolution of vulnerability and incidents are being effective and done in a desirable period of time.

4.3.3 TPerf

The metric TPerf, Team's performance, is a combination of the PETVI and ERVIDENT metrics. By having the combination of both metrics we can visualize what is the necessary effort to achieve the wanted efficacy. The goal is to have the minimum possible effort for a acceptable efficacy. This metric is in the Management category, Governance subcategory and its frequency of calculation should be monthly.

4.3.4 Trustworthiness blacklists' metrics

Number of reported incidents by month is a metric that counts the total number of opened and closed incidents for each month. This metric can be changed to just count the number of opened cases or closed cases. The metric can also generate sub-metrics for each type of incident (phishing, malicious attack, unauthorized access, etc.). The input data will be the incidents cases of that period, and their status (opened or closed). The output will be total number of reported incidents. This SM is inserted in the Process category, because it provides more management over the cases.

The Quiet Feeds metric calculates the number of feeds which are not providing any type of information. A feed can stop providing information due to its interruption or discontinuity of the information. With this information, the manager can reduce the number of untrusted feeds. The input will be the feeds and the information provided by each feed and the output will be the feeds or the number of feeds which are not giving information. This metric is inserted in the Technology category due its calculation in measuring the performance and compliance status of the metrics.

4.4 Visualization

It is highly recommended to use charts to summarize, represent and display the results of SM. The charts must be elegant, fast interpreted, and not too complex. Using the principles in the related work, we created three visualizations that provide qualitative and quantitative information, in a clean manner, for a better interpretation of the data.

The first visualization type is a line chart and represents the Team's performance metric (Sect. 4.3.3). The chart provides a visualization of the effort of the security team to achieve the degree of efficacy. This chart also provides the observation of the minimum effort required to achieve the wanted efficacy.

The second type is a combination of a bar chart and a line of progression. The visualization displays the outcome of the results which a cybersecurity tool is associated with and its precision, over the months. The bars represent the number of positive and false positive cases and the line is the precision. This visualization enables the comparison between the months, the precision of a cybersecurity tool's, providing quantifiable information to decide if it is reasonable to tune the cybersecurity tool or replace it by a more efficient one.

The final visualization type is a prototype and compares the results of the current month with the previous one, allowing a direct comparison between two months. We did not find anything similar in the related work about this visualization. However we can consider that our work is a variation of the work presented by Kotenko et al. [28]. Figure 4.2a and Fig. 4.2b display two examples of our idea. For both figures the outside circles represents the values of the previous month and the inside circles represents the values of the current month.

Figure 4.2a represents the number of opened and closed vulnerability cases. The right side represents the number of vulnerability cases that were opened and closed in the respective month, and the left side represent the total number of vulnerability cases that continued to be opened and the total number of vulnerability cases that are closed. A heat colour scale was used to represent the security status in a qualitative manner. In this example our heat was selected by comparison between the previous month values. The circle is green if the values are better than the previous month, i.e. as for the opened vulnerability cases is if the current month has a lower number of opened vulnerabilities than the previous month, and as for the closed is if the current month has a higher number of closed vulnerability cases than the previous month, yellow if are the same, and red otherwise. For the case of the total closed numbers the heat colour is always green because the number is always increasing. The outside rings heat colours represent the comparison between the previous month and its previous month. The SOC team can use thresholds to define the colour, instead of this colour scheme.

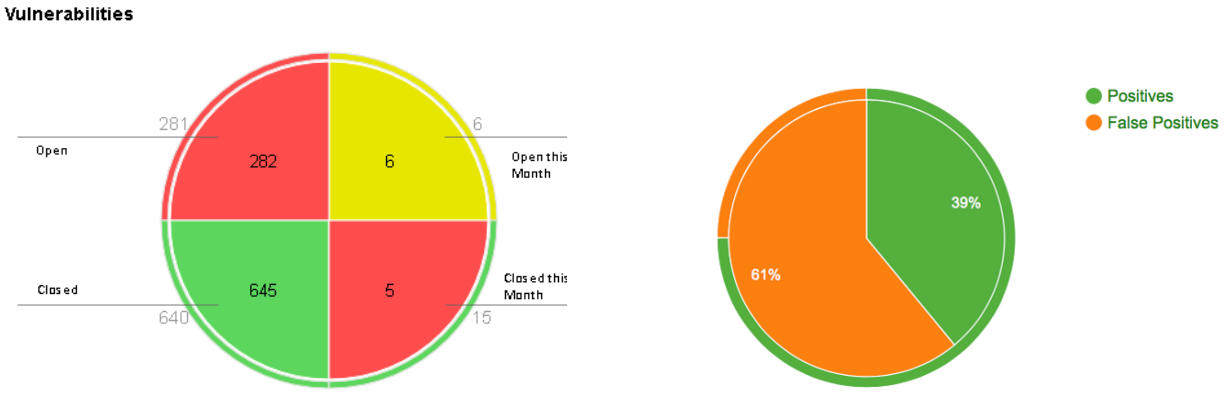
Figure 4.2b represents the number of cases which the result was positive versus the

number cases which the result was false positive. The green colour represents the true positive cases and the orange colour represents the false positive cases.

The goal is in an illustrative manner to verify if the number of false (or true) positives is higher, equal or lower comparing with the previous month.

For Fig. 4.2a and Fig. 4.2b the threshold values for the colours can, and should be modified by the C-level manager to be in accordance with the real security scenario of the organization.

The heat colours provide a fast information extraction about the state of the current month in comparison with the previous month.



(a) Status of the vulnerability cases

(b) Incidents' result

Figure 4.2: Visualization Prototypes

4.5 SM solutions

The solution for Security Metrics for SIEM systems is a set of well structured Security Metrics (Appendix A) and new visualizations models.

The purpose of the questionnaire and the survey is to know if the chosen SM are the appropriate for the C-level manager and SOC team. In order to know that, is necessary to understand which metrics each partner considers relevant, which metrics are already being used by the partners, and if it's possible to gather the required data, from the SIEM or other source, to produce the SM.

The visualizations types, in a qualitative manner, visualize the performance of the team in resolving the incidents, the efficacy of a tool by showing the precision over the past months and a comparison between the current and the previous month.

The second type of visualization is used in the Interface module (Sect. 5.1.4), and the two new visualization prototypes can be implements in a system in future work.

Chapter 5

Trustworthy Blacklists

"If you know yourself but not the enemy, for every victory gained you will also suffer a defeat.", [40]

This chapter describes the developed framework that gathers and assesses the quality of the public blacklists and their contents. The framework also allows to manage the blacklists and the incident cases associated with suspicious IP addresses from the blacklists, and produces a file containing the suspicious IP and their evaluation position. The created file is used by the SIEM rules to monitor the organization network and alert the SOC team when a suspicious communication occurs. The chapter starts by explaining the framework's architecture, and then describes each module, in more detail.

The information security teams, to enhance the security of the organization and to improve their defences by knowing the cyberthreats, are allocating their resources to obtain information from the Internet. However, we should have a measure of how reliable and trustworthy are the information provided by public sources.

A source, or a feed, is an entity that provides one or more lists. There are two types of lists: public and private. Public lists are unrestricted lists, i.e. can be accessed by anyone. On contrary, the private lists are restricted, i.e. only who has clearance can have access to the information. The lists which are more commonly used for cybersecurity are blacklists. A blacklist identifies suspicious hosts or malicious contents. In our work, we only consider public blacklists that contain information about IP addresses.

In addition of gathering information about cyberthreats, we want to determine how trustworthy are the public blacklists and the information they provide, regarding the correlation of information between the blacklists' content and the security incidents of the organization. The collection and the assessment should be autonomous, continuous and should consider the security status of the organization.

One objective of this work is the reduction of the number of false positive cases by alerts related to communications with IP addresses suspected of malicious activity, without reducing the number of positives cases. To reduce the number of false positives cases,

the quality of each blacklist and of their reported IP addresses are evaluated. The assessment of the trustworthiness considers information provided by the blacklists and the events of the organization regarding communications between the organization and an IP address suspected of malicious activity.

5.1 Architecture

The solution was developed to be used by all types of SOC teams (small, medium or big) and SIEM systems, and has four modules: IP Collector, Trustworthiness assessment, TABI Console, and SIEM rules. Each module is independent, i.e. the organization can implement the module that is suitable for their security goals, or use the all of them, as a solution, to have all the functionalities of gathering, assessing, monitoring and managing IP addresses and public blacklists. The first module is the IP Collector, a program whose purpose is gathering, filtering and normalizing information from public blacklists. The Trustworthiness assessment module evaluates the reputation of the malicious IP addresses and the trust of blacklists that contain them, considering information, internal and external, about the IP addresses and the blacklists. The Trust Assessment of Blacklists Interface (TABI) application consists of a centralized web management interface with management features and containing all the information about the IP addresses, blacklists and cases related with communications between the organization network and IP addresses suspicious of being malicious. Lastly, SIEM rules are defined to monitor the organization network and generate and alarm when a suspicious communication occurs. The SIEM rules, to monitor the suspicious IP addresses, use a reputable list of IP addresses (BADIP.csv).

Figure 5.1 illustrates the overview of the framework and presents the interactions between the modules.

5.1.1 Software Requirements & Database

Before explaining each component, is explained the languages used to create the framework, the dependencies and the required packages.

The IP Collector and the Trustworthiness assessment were written in python version 3. The IP Collector runs continuously, and the two programs require the installation of python version 3.

The Interface was written in JavaScript, HTML, PHP and CSS, under APACHE and PHP7. The requirement for the TABI is the installation of the APACHE and PHP (recommended version 7 or higher). The TABI source code contains a configuration file that is required to correctly configure and set the required fields to access the database.

The database chosen for this project is MariaDB version 10.0.29 [18], because it is a fully open source database, was created by the original developers of MySQL, has a

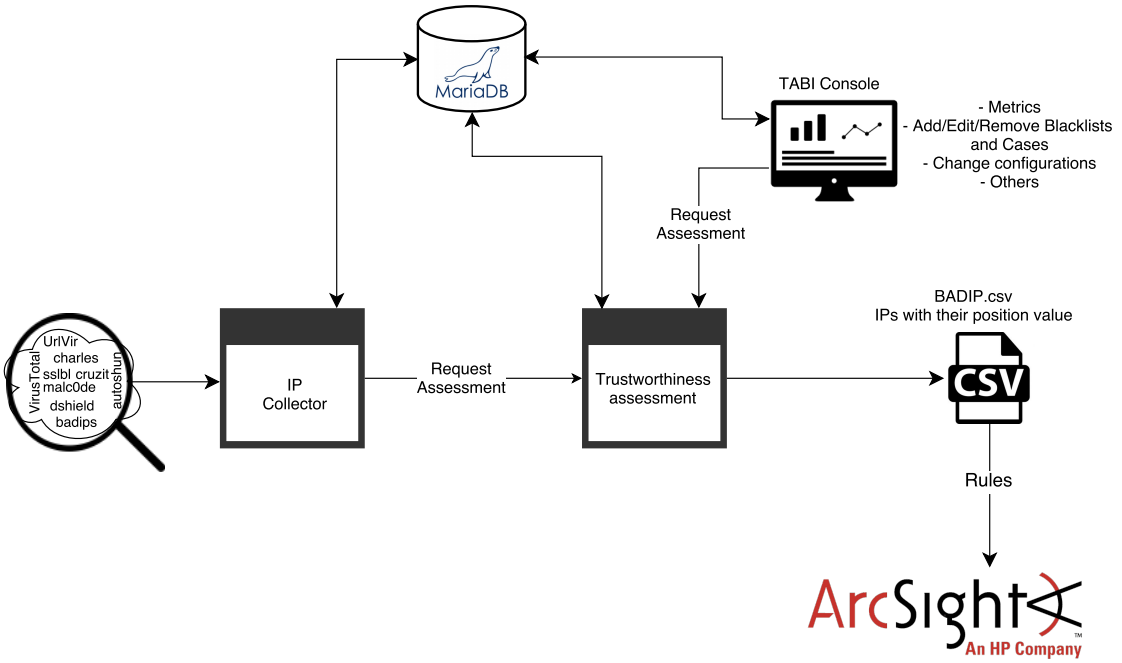


Figure 5.1: Workflow of the framework

high scalability with easy integration and additional storage engines. The database stores all the parameters used by the IP collector, Trustworthiness assessment and the TABI modules. The modules can read, write and modify the columns of the database's tables. Appendix C.1 has the UML model of our framework.

5.1.2 IP collector

The IP Collector module gathers, filters, normalizes, and categorizes information from a set of public blacklists, and runs continuously. The framework uses the OSINT concept to gather information of a preset of public blacklists. In the end of a period of two months of investigations, we selected 28 sources and 121 blacklists that were used from those sources. Appendix B presents all the blacklists with their URL and requisites. The blacklists were selected by the popularity factor, communities' reports and our own investigation.

The information is daily collected, and for each day the IP Collector adds the IP addresses which had not been previously collected. The data of each public blacklist is collected and is searched for valid IP addresses. If they exist, they are normalized and it is verified if they are already in the list under construction. For each IP address that is in the list but has not been reported yet by the blacklist under analysis, the number of occurrences of the IP is increased, is created a connection between the IP address and the public blacklist, and the variable of the last time the IP address was seen is updated with a new timestamp. If the IP address is not already in the list, it is inserted with value 1 in the number of occurrences, a connection to the current public blacklist is made and the first

time seen and last time seen variables will have the same timestamp. When this process is concluded, the IP Collector requests an assessment for all the blacklists and IP addresses. Afterwards, it sends the collected and evaluated data to the database to be updated.

Algorithm 1 describes the main process of collecting and processing information from public blacklists by the IP Collector.

The IP Collector and the Trustworthiness assessment programs use two main classes: Blacklist and IP, representing a public blacklist and an IP address, respectively. The two classes contain similar variables: *positives* and *false positives*, representing the number of positives and false positives cases which the blacklist or IP address is related to. The variable *historic*, for the blacklist case is an array with the score of the previous three months, the IP class is a Boolean array representing the persistence of an IP in the three previous months. Both contain a list with references for the association between them. In the case of the Blacklist class the lists contain the references for all the IP addresses which the blacklist reported and in the case of the IP class the list contains the references for all the blacklists which contained the IP address. The values of these attributes, with exception of the attribute *historic*, are for the current month. The Blacklist class has three extra attributes, the attribute *name* (as the blacklist's name), the attribute *url* (URL associated with the blacklist), and *apikey* (if the blacklist has an API Key associated is the API key value, otherwise will be set to null). The IP class contains more two attributes, the variable *address* and the attribute *occurrence*. The *address*, as the name suggests, is the address of the IP and *occurrence* is the number of blacklists which contain the IP in the current month.

Gathering, filtering, normalizing and assess the information

A few procedures are required before beginning the process of gathering and analysing the raw data from the blacklists. When the program is executed for the first time, it saves the current month (line 2 in the Algorithm 1), which is used to verify the transition between months. In each iteration of the loop, i.e. each day, the program begins by sending two requests to the database: one for retrieving information about the blacklists, and one for retrieving information about the IP addresses. The information request for blacklists (line 4) is: name, number of true and false positives, historic of the three previous months, IP addresses associated, URL associated, and API Key (if the blacklists does not need API key to retrieve the information, the API key variable will be null). A similar request is made to retrieve information about the IP addresses (line 5): address, number of true and false positives cases, historic, and blacklists associated.

After gathering this information, all associations between the blacklists and IP addresses are created. At this point, the program will have two lists, one for blacklists and another for IP. Then, it verifies if the month changed. If a month changed, the program calls the function *historic_update* (line 7). This function, as the name suggests, shifts and

Algorithm 1 IPs collector ▷ runs continuously

```
1:  function COLLECTOR()
2:      month = current month
3:      while true do                          ▷ for the program to run continuously
4:          blacklistsList ← all blacklists' information in database
5:          ips ← all IP addresses' information in database
6:          if month not equal to current month then
7:              historic_update(ips,blacklistsList)
8:              month = current month
9:          for each blacklist ∈ blacklistsList do
10:             raw_data = GETDATA(blacklist)
11:             for each line ∈ raw_data do
12:                 line_ips ← every IP present in the line
13:                 for each ip ∈ line_ips do
14:                     if isIP(ip) then
15:                         normalized_ip = NORMALIZE(ip)
16:                         if normalized_ip ∉ ips then
17:                             ADDIP(ip, ips, blacklist)
18:                         else
19:                             UPDATEIP(ip, ips, blacklist)   ▷ associate the blacklist
    with the IP
20:         ASSESSMENT()      ▷ trustworthiness assessment for the IPs and blacklists
21:         INSERTDB()
22:         seconds ← number of seconds for the next day
23:         sleep seconds
```

updates the historic of each blacklist and each IP address, for the month that has passed. In the IP address case, after shifting, the value, for the month that has passed, is set to true if the IP was reported by any blacklist, or if its precision value (described in the assessment subsection) is positive. Otherwise, it is set to false. After assessing the three months, all IP addresses without any value set to true will be discarded from the IP list and will be disassociated from the blacklists. The rest of the IP addresses will also be disassociated, and all information, with the exception of the *historic* and the *address*, is reset to zero. As for blacklists, everything is considered relevant, so we save all information in a data structure, specific to each blacklist. Then, as done for IP, every variable is reset, except the URL, the blacklist's name, and the API Key, followed by the removal of existing associations.

After this update, the program begins gathering information from each blacklist obtained from the database (from line 9 to line 19). This process extracts raw data via URL requests, using, when necessary, an API Key for the effect. This key is exclusive, and optional, for each blacklist. When analysing raw data, we use regular expressions to capture existing IP addresses. Then, we normalize each IP, so that every address follows the same syntax, which facilitates the comparison. As an example, consider the IP of blacklist A to

be 194.145.023.032, and the IP of blacklist B to be 194.145.23.32. Before normalization, these two are considered to be different IP addresses. However, by removing unnecessary zeros, both addresses follow the same syntax and, therefore, are considered equal.

After normalization, each IP is checked if it is in the under construction list. For each IP that is in the list but has not been reported by the blacklist under analysis, the reported number (*occurrence*) for this IP is incremented, the last time seen value is updated and a connection between the IP and the blacklist is created. If the IP is not already in the list, it is inserted in the list, with a value of 1 in the reported number, a connection between the IP and the blacklist is created, and the values of first time seen and last time seen are initialised with the current timestamp.

When the process of gathering the information of the public blacklists is finished, the IP collector request an assessment of trustworthiness for each IP address and blacklist (line 20). Upon the Assessment's completion, the new information is inserted into the and the IP's collector program sleeps X seconds to the next day to restart the process of gathering information from public blacklists. All the processes are repeated within the loop.

5.1.3 Trustworthiness Assessment

For a cyber defence, and when there is an extensive number of IP addresses of a blacklist or the result of collecting IP addresses from several blacklists, it's necessary to differentiate a suspicious IP address from another by their trustworthiness and reputation. The components used to differentiate are the criticality, credibility, impact, maliciousness and the number of reports. The trustworthiness assessment aims to classify the reputation of maliciousness of an IP address and the reputation of credibility of a blacklist considering these conditions.

Reputation of an IP address

To calculate the reputation of an IP address, four components are used: *tf*, $precision$, $average$ of the credibility of the blacklists that reported the IP and $persistence$. The calculus of the components, with exception of the component $persistence$, considers the values of the current month. The reputation, the $precision$ and the other metrics values are between [0;100].

The term frequency, tf component, is the relative frequency of the IP comparatively with the maximum IP occurrence on that month, and is calculated by dividing the number of occurrences of the IP i (the number of blacklists that reported the IP i) by the maximum number of occurrences (5.1), thus having a relation of relevance of the IP address.

$$tf_i = \frac{IP_{Occ_i}}{max_{(k=1,\ldots,n)}(IP_Occ_k)} \times 100 \tag{5.1}$$

Where,

- $IP_{Occ_i} \leftarrow$ the number of lists that the IP_i appears in the current month.

The $precision$ is the component which uses the internal knowledge of SOC to tune the trustworthiness of an IP. The component considers the investigation of suspicious IP addresses related with incident cases and its maliciousness in the organization to asses the IP address. The $precision$ of IP i is the ratio between the number of confirmed cases of malware detected by communications with the IP i and the total number of cases associated with communications with that IP in the current month, i.e. Positive cases and false positives cases (5.2).

$$precision_i = \frac{Positives_i}{Positives_i + FalsePositives_i} \times 100 \tag{5.2}$$

Where,

- $Positives_i \leftarrow$ the total number of confirmed malware detections for the IP address i;

- $FalsePositives_i \leftarrow$ the total number of incidents, associated with IP i, for which the performed investigation did not found malware presence in the asset.

The $persistence$ of an IP i is defined for a period of three months and is a measure of the IP permanence throughout this period in blacklists or in positive cases. As one of the objectives of our work is to adapt the program to the environment of the organization, when an IP has not been informed by blacklists, it is only discarded if it is not associated with positive cases (precision is zero). We consider the current month to be month 0. Thus, the previous three months are negatively indexed with -1, -2, and -3, respectively.

The $persistence$'s value is obtained by a weighted sum of the IP's permanence over the months, as given by (5.3).

$$persistence_i = \sum_{k=-3}^{-1} (month_{k_i} \times weight_k) \times 100 \tag{5.3}$$

Where,

- $Month_{k_i}$ = 1, if the IP i was present in month k. 0, otherwise, k = -3,-2,-1

- $Weight_{-1}$,$Weight_{-2}$,$Weight_{-3}$, are parameters to weight the persistence over the previous months and their values are $\frac{1}{2}$, $\frac{1}{3}$, and $\frac{1}{6}$, respectively.

As explained in Sect. 5.1.2, at the beginning of each month it is verified if the IP was reported by a blacklist or if its precision is higher than zero. The weights were adapted by computational experience so that maximum sum is 1, the minimum is 0 and the recent previous months have more weight than the older. Table 5.1 presents the possible values of persistence according to the presence of or not of the IP in the previous three months.

$(\frac{1}{2})\ mont_{-1}$	$(\frac{1}{3})\ month_{-2}$	$(\frac{1}{6})\ month_{-3}$	weight's sum	$persistence$
✓	✓	✓	1,00	100,00
✓	✓		0,8(3)	83,3(3)
✓		✓	0,6(6)	66,6(6)
	✓	✓	0,50	50,00
✓			0,50	50,00
	✓		0,3(3)	33,3(3)
		✓	0,10	10,00
			0,00	0,00

Table 5.1: Combinations of possible presence and the IP's persistence values

The $average$ component is the average trust score (see next section) of the blacklist values, that reported the IP address. The $average$ component is used to differentiate the IP sources. If blacklist A is more trustworthy than blacklist B, then the IP addresses reported by blacklist A should have more importance in credibility than the one reported by blacklist B.

The reputation of an IP address is trustworthiness, after the assessment, in the maliciousness of the IP address i, and is calculated by the average of the sum of all the four components.

$$IPReputation_i = \frac{tf_i + persistence_i + precision_i + average_i}{4} \tag{5.4}$$

Where,

- $average_i \leftarrow$ is the average trust score of the blacklists associated with IP i.

We add a new equation to sort the IP addresses by positions. This new equation was created to aid the SIEM rule when selecting the suspicious IP addresses to be monitored. In our investigation and analysis we observed that the Trustworthiness assessment could give a small dispersion of the values and the maximum reputation of an IP address could be small or different over the months. Therefore, to create a standard rule that could be used for all months without changing its parameters we created this new metric which calculates the position of IP i by comparing its reputation value with the maximum reputation value calculated in the month, as observed on (5.5).The $IPPosition$ metric value is between [1;100], and one or more IP addresses can have the same position.

$$IPPosition = \frac{IPReputation_i}{max_{(k=1,..,n)}(IPReputation_k)} \times 100 \tag{5.5}$$

Trustworthiness of a blacklist

For the calculation of the trustworthiness of a blacklist, the components used are: $rank$, $precision$, $history$. The trustworthiness score, the $precision$ and $history$ values are between [0;100].

The $precision$ of a blacklist j is the ratio between the number of confirmed cases of malware detection caused by communications with the IP addresses reported by the blacklist j and the total number of cases associated with blacklist j, in the current month, as inferred by the result of (5.6).

$$precision_j = \frac{Positives_j}{Positives_j + FalsePositives_j} \times 100 \tag{5.6}$$

Where,

- $Positives_j \leftarrow$ the total number of confirmed malware detections for IP addresses in blacklist j in that month;
- $FalsePositives_j \leftarrow$ the total number of incidents, associated with IP addresses of blacklist j,for which the performed investigation did not found malware.

The $history$ component is intended to evaluate the reputation of the blacklist j over time. This component is defined by the sum of the blacklist j reputation value of the last trimester, with each month having a weight associated, as postulated by (5.7).

$$history_j = \sum_{k=-3}^{-1} (month_{k_j} weight_k) \times 100 \tag{5.7}$$

Where,

- Each variable $month_{k_j}$ has the score's value of the blacklist j in month k, k=-3,-2,-1,
- $weight_{-1}$, $weight_{-2}$ and $weight_{-3}$ are parameters to weight the score of blacklists over previous months and their values are $\frac{1}{2}$, $\frac{1}{3}$ and $\frac{1}{6}$, respectively.

The trustworthiness of blacklist j is the average of the components: $precision_j$ and $history_j$;

$$BlacklistTrust_j = \frac{precision_j + history_j}{2} \tag{5.8}$$

Alternatively, we could have used different weights for the components.

The assessment program is written in python and is called by two components: the IP Collector and TABI. When the IP Collector completes the gathering process, it calls the assessment program to evaluate the trustworthiness of the blacklists and the IP addresses. Whenever there is an update of the status of an incident cases, i.e. true or false positive, the assessment program is called to recalculate the *IPPosition* of each IP address and the trust of each blacklist.

After the calculation of the trustworthiness of the Blacklists and the IP addresses, the assessment program updates the database's information and re-writes the file 'BADIP.csv'.

5.1.4 Trustworthy Assessment Blacklists Interface

The Trustworthy Assessment Blacklist Interface (TABI) is a web interface created to manage and visualize information related with blacklists, suspicious IP addresses, incidents and public IP addresses of the organization. TABI allows centralized management of the entire framework, without the need of writing code, and presents valuable information about the status of the framework. The application allows the addition, removal and edition of blacklists and incidents, to be used in the trustworthiness assessment of the IP addresses and blacklists. The TABI application has an extra functionality that indicates if a public IP of the organization is reported by any public blacklist in the database. For this functionality to be operational is required to add the public IP addresses of the organization into the database through TABI.

TABI uses an architectural model, Model-View-Controller (MVC), using CodeIgniter [15], and was written using web languages, such as HTML, PHP, CSS, and Javascript.

Although TABI offers various functionalities and displays several SM throughout its pages, this section explains only some of the displayed SM and basic functionalities.

The homepage of TABI contains an overview of the solutions and organization status (Fig. 5.2): 1) The *# of IP* represents the number of IP addresses collected from the public blacklists; 2) The percentage of *Quiet Blacklists* indicates the percentage of blacklist from which no IP address was reported; 3) *# of Cases* represents the number of open cases related with suspicious communications with one or more suspicious IP addresses; 4) *# Organization's IP* is the number of public IP addresses belonging to the organization that are in the suspicious IP list. If an asset of the organization is infected, it can perform perform malicious activities and be reported by other organizations or web communities. If none of the public IP addresses of the organization is referenced by any blacklist, the word *Clean* is displayed.

Figure 5.2: SM displayed in homepage of TABI - 1

In addition to these four SM, the homepage displays additionally three SM related with the top 10 malicious IP addresses and trustworthy blacklists, ordered by score, and the last 10 opened cases, as illustrated by Fig. 5.3 when comparing Fig. 5.2 and Fig. 5.3,

we see that in the last 10 cases there are two cases and in the *# of Cases* of Fig. 5.2 only indicates one case. This is because we wanted to show that in the *# of Cases* only presents the opened cases. When any of the displayed items is pressed, TABI redirects the user to another page with more details about the pressed item (suspicious IP, public blacklist, or Case).

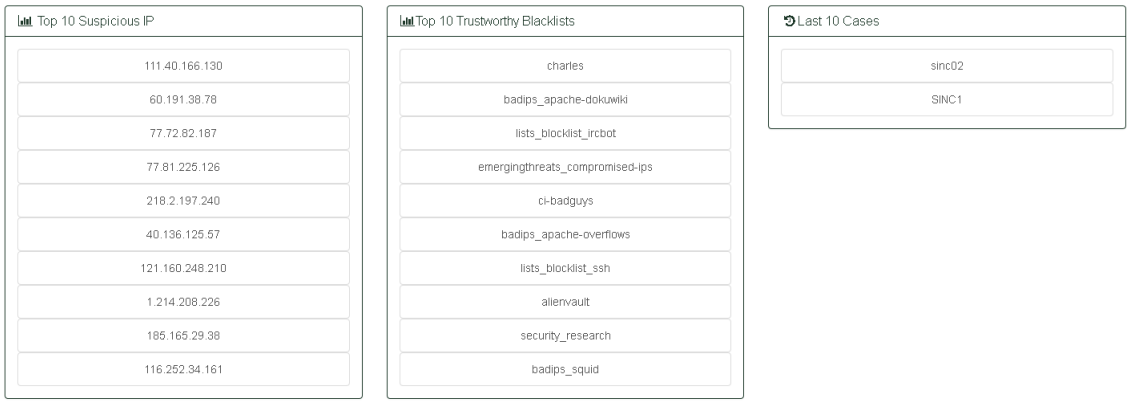


Figure 5.3: SM displayed in homepage of TABI - 2

These generic metrics provide to the SOC a comprehensive understanding about the security state of the organization and the health state of the framework.

TABI provides the functionality of adding, editing and removing a blacklist, a case and public IP of the organization. It is possible to add one or more of these three elements. Although, for both adding options is necessary to provide some required inputs, such as a name to represent the new item. To edit or remove an element is necessary to go the webpage of the element. When pressed the edit button, a dialogue box appears with all the editable fields. As for the case of remove button a confirmation dialogue box appears.

Besides the information provided in the main webpage, TABI offers additional pages for the four main element: Blacklists, suspicious IP, Organization's Cases and Organization's Public IP. Each page has a filtered table which contains all the items from the database. In the page of suspicious IP, due to the large quantity of IP gathered, only 500 IP addresses are displayed. However, if the user wants to see them all, there is a button to display all the IP addresses from the database in a csv file (BADIP.csv).

Each element has its own webpage containing all its information. For example, in the webpage of a blacklist a precision table is displayed to observe the precision of the blacklist over months, this visualization is described in Sect. 4.4. Figure 5.4 illustrates the precision of a blacklist over months. Also all the information available in the database related with the blacklist is displayed in a table.

In future work, we expect to implement more manageable functionalities and metrics in the TABI module, for example functionalities to control the IP collector module and metrics that illustrate information about the organizations most affective sectors.

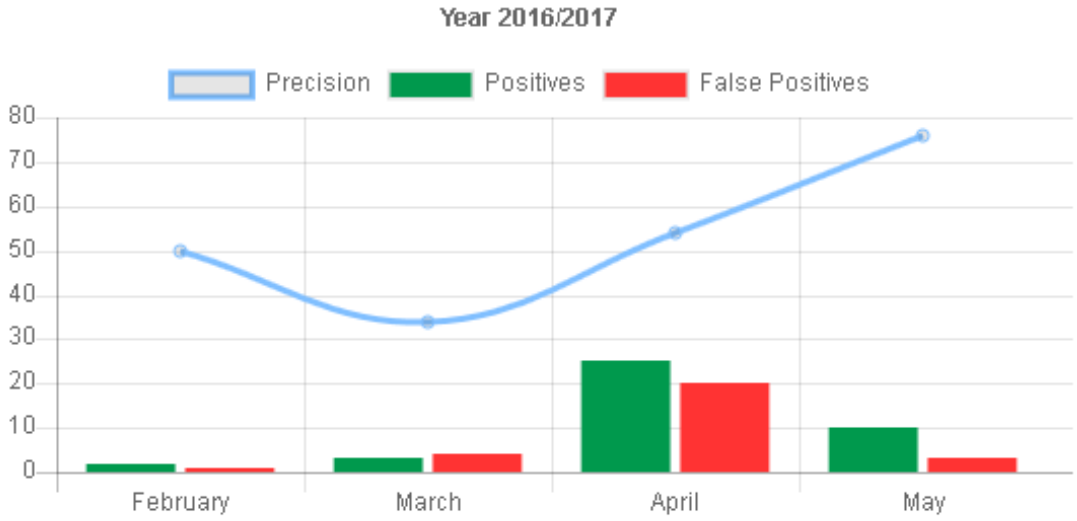


Figure 5.4: The public blacklist's precision over months (example)

5.2 SIEM

The final phase is the incorporation of the appraisal results into the SIEM infrastructure, by creating SIEM rules to monitor the security events and alert when communications between the organization network and the assessed suspicious IP addresses occur. Currently, we are focused in the detection of the organization's assets with malware infections. Regarding that, we want to investigate the communications which started in the organization and the target is one of the suspicious IP gathered in the previous phase. These events are from logs of several devices. To implement this into the SIEM, is necessary: the BADIP list, the SIEM rules, and the SIEM sources.

5.2.1 BADIP list

The BADIP list, is a CSV file which contains two columns. The first column contains the suspicious IP addresses and the second column contains their position value (5.5), values between 0 and 100. Every time the assessment module is called; this file is updated. This file is the source used by the SIEM system. The rules created in the SIEM read the file and select the IP addresses with a position in correspond to the values defined in the rules. This file can be accessed through the interface TABI to be used by third-party products.

5.2.2 SIEM rules

A SIEM rule is a real-time component that evaluates input events for specific conditions and patterns. When an event occurs for which a rule matches, an action is triggered

in response. Rules trigger automated actions or alerts to the SOC team to analyse and monitor specific type of events. To monitor and alert communications with suspected malicious IP addresses, it is necessary to create rules in the SIEM system that alert when an organization IP address performs one or more communications with one or more IP addresses of the list.

Triggering a rule depends on the tuning of several parameters: the time interval observation, the number of communications in that time interval, and the number of different IP addresses suspected in those communications. The definition of these parameters should consider the trustworthiness level of each IP to reduce the number of false positives and to have a greater confidence in the alarms triggered by these rules.

For our case, the rules not only will be used to monitor and alert, but they are tuned to aid in the reduction false positives cases. Whenever the assessment performs a change in the BADIP file, the SIEM rule will collect each set of IP addresses that contain the position value range that the rule is set to handle, and monitor the organization's network to alert when there is a communication between the organization and those IP addresses.

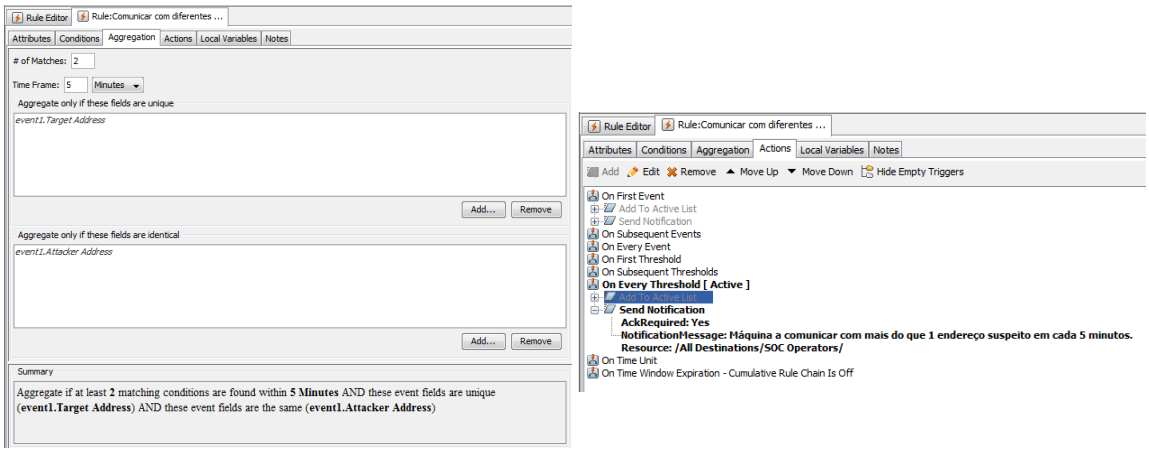
We create one prototype rule to select the IP addresses which position is higher than 84, and to alert when one communication starting from the organization and destination as one of the selected IP. The options for this rule are displayed in the Fig. 5.5a and Fig. 5.5b. We used a dynamic list and in the section condition we associate the list with the rule. In a time frame of five minutes we wanted to have two event matches. These two events must contain the same attacker, the organization host, and different target addresses, i.e. the organization host must communicate with two different malicious hosts to trigger the rule. We defined these configurations because the first configuration rule was with only one communication and started to overload the system with alert notifications.

An additional rule must be created to select the IP addresses with position equal or higher of 85. These rule, the configuration rule, will select the dynamic list and select the element of the first column, i.e. the IP addresses, which have a value equal or higher than 85 in the second column, i.e. the position value. After selecting the rule will create a new dynamic list which it will be used by our prototype rule.

5.2.3 SIEM Sources

As formerly described in Sect. 2.3, multiple sources can feed the SIEM with security information, the only requirement is the source's connection with a SIEM connector. The connectors must recognize the syntax provided by the source, process it and send it to the Logger and/or ESM.

Throughout this work, log sources were added to help the rules created to capture the suspicious communications, detect the infected asset, following by the analyze of the asset for malware infections. In the end, six sources were used: firewall, IPS, WAF, VPN, DHCP and antivirus.



(a) Attributes option (b) Condition option

Figure 5.5: SIEM rule configuration options

Firewalls supply logs related with communications between devices (Internet or inside the organization). To provide information to the Trustworthy blacklists framework, the events of the firewall must contain the destination IP (target), the source IP (attacker) and the action made by the firewall.

The Intrusion Prevention System (IPS) logs are identical to the logs of the Firewall. They also supply the SIEM with communications logs between the organization and the Internet. The difference is the logs of the IPS are about intrusion events. It contains the destination IP (target), the source IP (attacker) and the event type (can be a connection allowed or the name of a malicious signature).

For more information, such as the requested URL and the user name (who made the communication) the logs of the Web Application Firewall (WAF) can be used. In addition of providing the destination IP (target), the IP associated with the domain, and source IP (attacker), WAF logs additionally provides the hostname, username (the user that was in that session with that IP at that moment), the name of the event and the request URL. The requested URL can be valuable information to verify if the IP that is considered malicious is associated with a malicious URL. Besides, this information can be used to associate the malicious IPs with malicious URLs and reduce the false positive rate.

Virtual Private Network (VPN) logs are related with remote access and can be used to obtain the hostname and username from the attacker and target. These logs can also provide the source's and destination's IP, and the action performed.

DHCP relates IP addresses with hostnames. The information provided by the DHCP appliance assist the SOC team to identify the machine suspected of infection to be analysed.

The logs of the Antivirus are used in the last phase of the process, confirming if the alarm made was a true positive or false positive. With the result of the confirmation the Trustworthy Blacklist will re-calculate the score of the IP's reputation (5.4).

Chapter 6

Results

In December of 2016, because of this work - gathering information from public lists and classify their content in the matter of trustworthiness, considering the internal information - EDP's SOC engaged in the process of gathering information from public and private blacklists and started to analyse the communications between the organization's assets and suspicious IP addresses. After the analysis, if it is possible to find the asset, an incident case is created to diagnose the asset for infections and eliminate the malware, if found. In the end of diagnosis and cleaning process, each incident case is closed and classified as True Positive (TP) or False Positive (FP). A TP case is a case which there was confirmation of malware detection in the asset related with the case. A FP case is for which there is no evidence of malware presence in the asset related with the case. In addition of the new task, the SOC created a public-private list containing IP addresses from public and private blacklists, with a high trustworthiness reputation from cybersecurity communities, and IP addresses from internal analysis which the SOC considered malicious. This public-private list here on after will be referenced as OSINT-LIST.

SIEM rules were also created to alert whenever occurs a communication with origin within the organization network and with destination to one or more IP addresses from the list. This condition is important for the analysis because when investigating the events associated with the cases the target is the malicious IP and the attacker is the organization infected asset. An incident case can be associated with one or more suspicious IP addresses, as result of the events aggregation capability of the SIEM.

To increase even more the SIEM detection of malware infection, the SOC team uses cybersecurity appliances, such as WAFs, Firewalls, IPS, IDS, private lists provided by the technical support of the cybersecurity appliances, and more. These tools monitor and prevent malicious communications between the organization and suspicious IP addresses.

The following sections describe the practical case study and our observations between our list, BADIP list, and the two lists used by SOC team of EDP accordingly with the obtained results.

6.1 Preparation & Practical Case Study

We selected 121 public blacklists to retrieve and assess their content. Our list, BADIP, covers blacklists which are already considered trustworthy from the communities, for example RansomwareTracker and Virustotal, and others for which there is no information available about their trust, such as Charles and CryptoPHPMaster. After the conclusion of the IP Collector process, the BADIP list had per month, an average of 166750 IP addresses. If we inserted all these IP into the SIEM, the system could became overloaded and the incident cases created could be too excessive to be handled by the SOC and other involved teams. While a solution was being thought and tested, each month a manual analysis was performed to compare the incidents cases of EDP generated by its two lists. We submit our list to the cases of the EDP's OSINT-list, in this instance we call our list BADIP list. And we submit our list to the total cases of the organization's communications with suspicious IP addresses, the ArcSight Global list. For this instance we call our list BADIPPotencial.

The ArcSight Global list is the instance that contains all the incidents associated with the alerts from all the paid appliances, including the OSINT-LIST, used by the SOC team to monitor and prevent suspicious network communications. These alerts are triggered from lists of IP addresses, network behaviour analysis, signatures, etc.

Because our list was compared with instances of cases of the other two lists, besides the true and false positives results we can determinate the true and false negatives. We use these two new components to calculate the *Accuracy* of our list, for the two instances. We consider *Accuracy* as a metric to know how precise our list is, in terms of classifying an IP as malicious or not malicious. The *Accuracy* is calculated by the sum of true positives and true negatives, divided by the sum of all four classification types (true positives, false positives, true negatives, and false negatives), i.e. all the cases of the instance. In conclusion, for the analysis, are designated four classification types for the BADIP Cases.

A case is *True Positive* (TP) if the original case (ArcSight or OSINT-LIST Cases) was positive and the BADIP list contained the malicious IP addresses.

A case is *False Positive* (FP) if the original case (ArcSight or OSINT-LIST Cases) was false positive and the BADIP list contained the suspicious IP addresses.

A case is *True Negative* (TN) if the original case (ArcSight or OSINT-LIST Cases) was false positive and the BADIP list did not contained the suspicious IP addresses.

A case is *False Negative* (FN) if the original case (ArcSight or OSINT-LIST Cases) was positive and the BADIP list did not contain the suspicious IP addresses.

As illustrated in Fig. 6.1 the workflow of our investigation consisted of for each case opened we did an investigation to extract the target IP. If one of the target IP addresses related with the incident case is public, the IP address is searched through the BADIP list to find if the list considers the target IP as malicious, and the incident cases are monitored

for the investigation conclusion to know if they are true or false positive. After the case being close we will determine the classification for our list for that case.

Our objective is to assess the efficacy of our solution, tune the framework, and to comprehend the behaviour of a public blacklist and suspicious IP addresses in an organization environment.

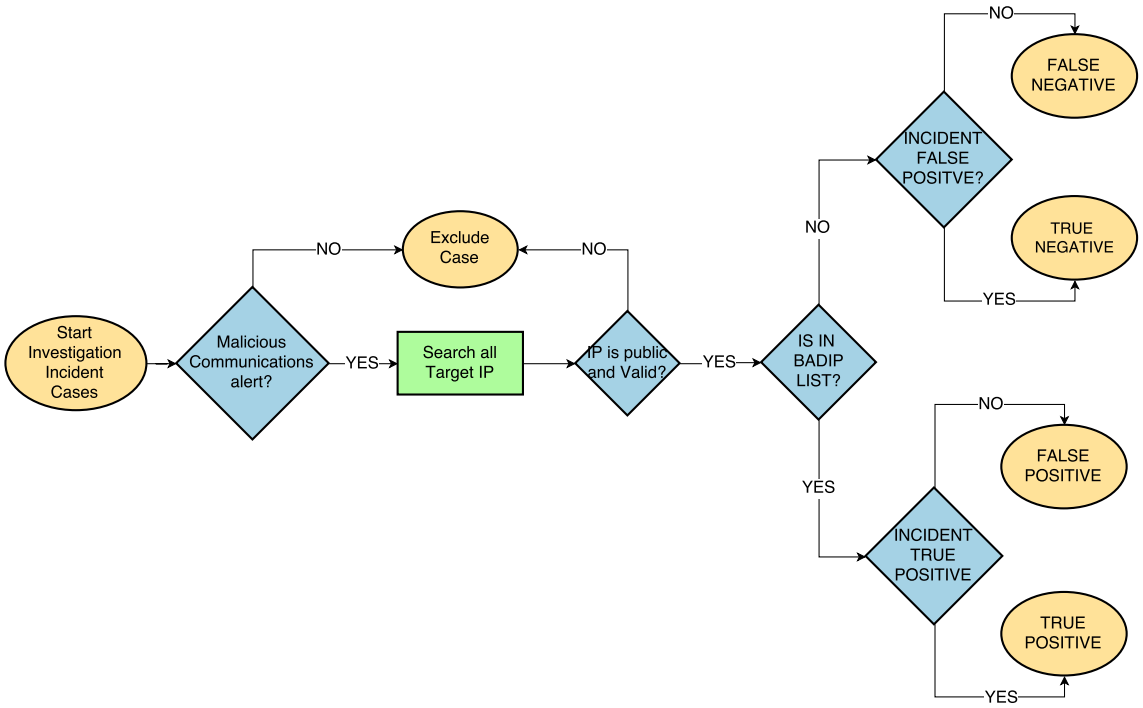


Figure 6.1: Workflow of the Case study analysis

6.2 Analysis and Results

The study was carried out over a five month period, from December to April, the IP Collector and the Trustworthiness assessment modules were analysed and tuned. The IP Collector was improved to gather efficiently the blacklists content, without discarding suspicious IP addresses, reducing the resources and reduce the required time to gather, filter, normalize, and correlate the information from the public blacklists. The equations of the Trustworthiness assessment were tuned to be representative of an organization's reality.

Our objective was to analyse and compare the BADIP list with the organization's lists. Analyse the presence and trustworthiness of the public blacklist. Tune and analyse the results of the Trustworthiness Assessment. The following subsections describe each of these studies.

6.2.1 Lists

In the list comparison we observe the efficiency of the lists in providing trustworthy information to detect suspicious communications within the organization's network. Table 6.1 presents the number of true and false positives cases, true and false negative cases, the total number of cases, the precision, the False Discovery Rate (FDR), and accuracy, per month and per each list. The FDR metric considers the TP cases and FP cases of a list and calculates the percentage of incorrect assessment presumptions in considering an IP address as malicious, i.e. of all the list associated cases the percentage of cases created due to the list's wrong identification of an IP address being malicious. The FDR is the total number of false positive cases divided by the sum of the total positive and false positive cases.

The ArcSight Global Cases is the number of cases related with the all the cybersecurity tool used by the SOC team to monitor and alert suspicious communications. The OSINT-LIST Cases is the number of cases resulting on the alerts of the public-private SOC's list. The BADIP Cases is the number of cases which the BADIP list could cover over the instance of cases of the OSINT-LIST. The Potential extra Cases, BADIPPotential, is the number of cases which the BADIP list could cover over the instance of cases of the ArcSight Global.

In the BADIP Cases line of the Tab. 6.1 is presented the number of *True Positives* (TP) cases, the number of *False Positives* (FP) cases, the number of *True Negatives* (TN) cases and the number of *False Negative* (FN) cases. Additionally, is displayed the total number of cases (the sum of the previous four components) and three percentages values: *Precision* (defined in Sect. 5.1.3), FDR metric, and *Accuracy*. The same components are displayed for the Potential extra Cases list. As for the ArcSight Cases and OSINT-LIST Cases fewer values are presented. As explained for the ArcSight Cases and OSINT-LIST Cases is not possible to calculate in a precise and accurate manner the True Negatives and False Negatives values, thus the table does not display these values, and therefore, due to the dependency of the *Accuracy* on these two components, the *Accuracy* is also not displayed - represented by the hyphen character.

From this preliminary study about the lists, we can observe that the difference of precision between our list and the organization's lists is more considerable in the month of December, and then stabilized over time. However, we should take in consideration for our evaluation that, due to this project, the organization tuned their list over the months to reduce the false positives.

By comparing our list with the two other lists we can observe that our list covered three cases, in January, that the OSINT-LIST did not cover. One in the month of February, and eight cases in April. This detection improvement is because of the persistence and precision metric. These results strengthen our hypothesis that we should use persistence and precision in their evaluation, and considering the analysis of the organization's cases

Lists	TP	FP	TN	FN	Total	Precision	FDR	Accuracy
December								
ArcSight Global Cases	23	12	-	-	35	65,71%	34,3%	-
OSINT-LIST Cases	11	5	-	-	16	68,75%	31,3%	-
BADIP Cases	5	1	4	6	16	83,33%	16,7%	56,3%
Potential extra Cases	5	2	6	22	35	71,43%	28,6%	31,4%
January								
ArcSight Global Cases	25	30	-	-	55	45,45%	54,5%	-
OSINT-LIST Cases	10	26	-	-	36	27,78%	72,2%	-
BADIP Cases	7	16	10	3	36	30,43%	69,6%	47,2%
Potential extra Cases	9	17	13	16	55	34,62%	65,4%	40,0%
February								
ArcSight Global Cases	25	8	-	-	33	75,76%	24,2%	-
OSINT-LIST Cases	12	4	-	-	16	75,00%	25,0%	-
BADIP Cases	7	2	2	5	16	77,78%	22,2%	56,3%
Potential extra Cases	8	2	6	18	34	80,00%	20,0%	41,2%
March								
ArcSight Global Cases	52	25	-	-	77	67,53%	32,5%	-
OSINT-LIST Cases	14	7	-	-	21	66,67%	33,3%	-
BADIP Cases	12	6	1	2	21	66,67%	33,3%	61,9%
Potential extra Cases	12	6	23	36	77	66,67%	33,3%	45,5%
April								
ArcSight Global Cases	25	19	-	-	44	56,82%	43,2%	-
OSINT-LIST Cases	4	4	-	-	8	50,00%	50,0%	-
BADIP Cases	3	4	0	1	8	42,86%	57,1%	37,5%
Potential extra Cases	11	10	9	14	44	52,38%	47,6%	45,5%

Table 6.1: Comparison between the lists values of the results of the cases

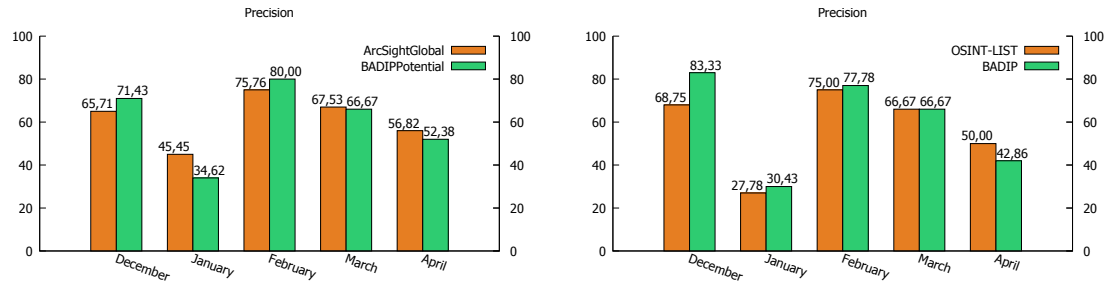
performed by the SOC team.

The component persistence not only uses the information of the public blacklists to determine if an IP address should persist in our list, but also verifies if the IP address precision is higher than zero. The precision is the component which evaluates an IP address and a blacklist by the information of the internal investigation of the SOC team. These two metrics are described, in more detail, in Sect. 5.1.3.

The list's precision, over the five months period, is depicted in Fig. 6.2a and Fig. 6.2b.

Figure 6.2a exhibits the precision values obtained for the ArcSight Global list and the BADIP list (here called as BADIPPotential), from December 2016 to April 2017. The ArcSight Global list had three months (January, March and April) with a higher precision against two months (December and February), which the BADIP list had a higher precision. March was the month that the precision between the two list were closer, only differ by 0,86%.

The average precision of the ArcSight Global list over the five months is 62,26%. The BADIP list precision for the same period and for the analysis the results of ArcSight Global list is 61,02%. The BADIP list had a percentage of 1,24 lower comparatively with the precision of the ArcSight Global list'. However, we are comparing a list, BADIP, that gathers information from public sources, with a list that obtains its information about suspicious communications from paid cybersecurity appliances, public-private lists and information manually inserted by the SOC team.



(a) Precision comparison between ArcSight Global and BADIP Potential list

(b) Precision comparison between OSINT-LIST and BADIP list

Figure 6.2: Comparison of the precision between the lists over December 2016 to April 2017

In the evaluation between the OSINT-LIST list and BADIP list, the lists that use blacklists, the results are propitious to the BADIP list. The BADIP only retrieves information from public blacklists and the OSINT-LIST retrieves information from public and private blacklists, and the SOC team introduced IP addresses that after their analysis they considered malicious. As demonstrated in Fig. 6.2b the BADIP list has a higher precision percentage. Only in the last two months, March and April, that the precision was equal and the BADIP had a lower precision, respectively. The BADIP list and the OSINT-LIST, in average, had precision of 60,21% and 57,64%, respectively. This means that our list had 2,57% better precision than the OSINT-LIST.

The other analysis of the BADIP regards the accuracy between the two instances that the BADIP list was under evaluation (Fig 6.3). Despite the fact that the Potential BADIP (BADIP-ArcSightGlobal in the Fig. 6.3) list covered more cases of the organization in the ArcSight Global list environment, its accuracy was lower than that of the BADIP list

in the environment of the OSINT-LIST. The average accuracy for the first environment (BADIP Potential in ArcSight Global cases) is 40,72% and for the second (BADIP in OSINT-LIST cases) is 51,84%.

Although our list (BADIP) had a lower average precision when compared with the ArcSight Global Cases (a list with blacklists and paid cybersecurity appliances), the BADIP list had a higher average precision when we compare BADIP and OSINT-LIST lists, which gather information from blacklists (public and/or private). Another factor is the difference between the averages. The BADIP Potential average precision is 0,86% lower than the ArcSight Global list. As for the OSINT-LIST, the BADIP list has a precision increase of 2,57%.

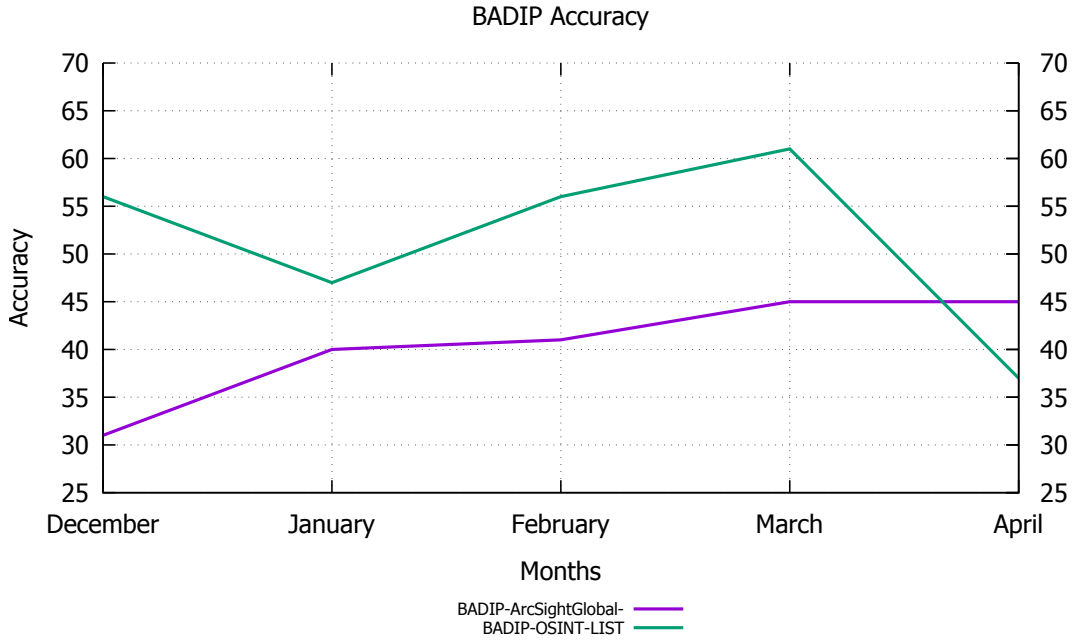
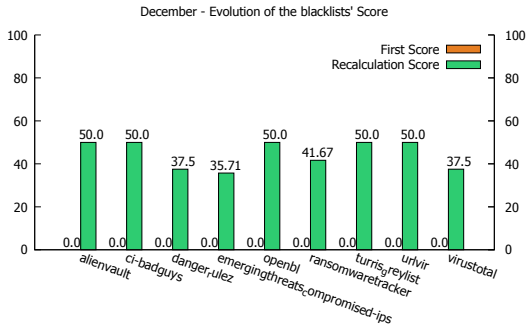


Figure 6.3: BADIP Accuracy in the two scenarios

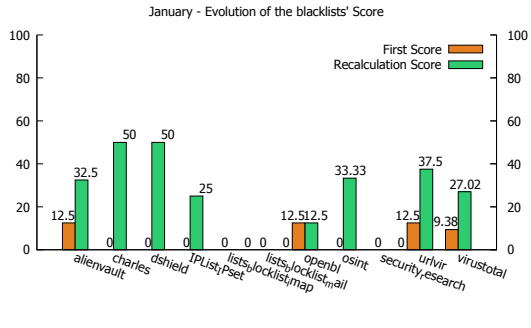
6.2.2 Analysis of the public Blacklists

The trustworthiness module assess the blacklists, accordingly with the results of the cases that the blacklists are associated with. The objective of this assessment is to prioritize the IP addresses which were reported by the blacklists that our solution considers more trustworthy, regarding their case history.

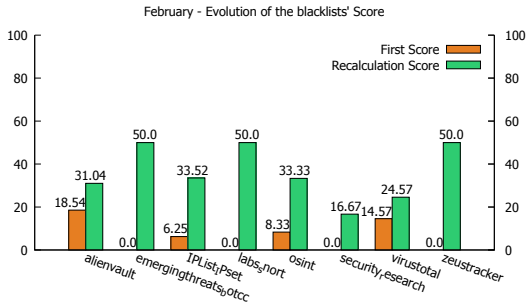
Figure 6.4 contains the analysis of the blacklists in each month of the study period. Each figure only contains the blacklists that were associated with cases in each month, and illustrates the initial trustworthiness score of the blacklist, orange colour, and the final trustworthiness score of the blacklist, green colour.



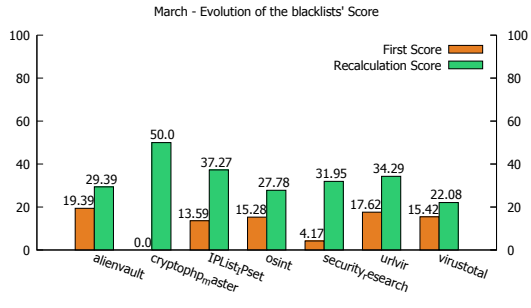
(a) Blacklists Trustworthiness Assessment in December



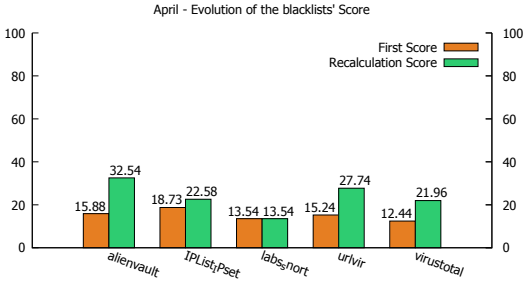
(b) Blacklists Trustworthiness Assessment in January



(c) Blacklists Trustworthiness Assessment in February



(d) Blacklists Trustworthiness Assessment in March



(e) Blacklists Trustworthiness Assessment in April

Figure 6.4: Blacklists initial and final trustworthiness score in each month

The blacklists which have an association with a case in all five months are the Alienvault and the Virustotal. We can also consider the IPlistIPset as one blacklist which was present in all months, because in December, the only month which the blacklist did not appear, the blacklist wasn't in the list of blacklists to retrieve information from.

One aspect that we observe is that the assessment module is usually giving more credibility the blacklists which do not have historic. In the first month most of the blacklist

have a precision of 100[1], but due to the (5.8), the final score is 50% of trust. The results of the study demonstrate that a blacklist with no history have more trustworthiness that the blacklist with historic, however if we analyse each case of blacklists with historic, these blacklists have false positive cases, therefore their precision decrease, and the blacklists with no history have a precision of 100%, because only contain one or two cases and they are positive. Our study confirms that the blacklists with history cover more cases, and therefore have more false positives than the blacklists with no history.

We see that in the initial months there is a significance difference of the trust score between the blacklists, but over the months this difference decreases. The blacklist starts to have a history and that influences the trust of the blacklist. Figure 6.5 illustrates over the five month period (December 2016 to April 2017) the score of the blacklists which were related with an organization case. And from there the normalization of the trustworthiness values is clear. From Fig. 6.5 we verify that January was the month with more public blacklist associated with cases, and April with the lower number of public blacklists and with the minimum difference values between the blacklist trust score.

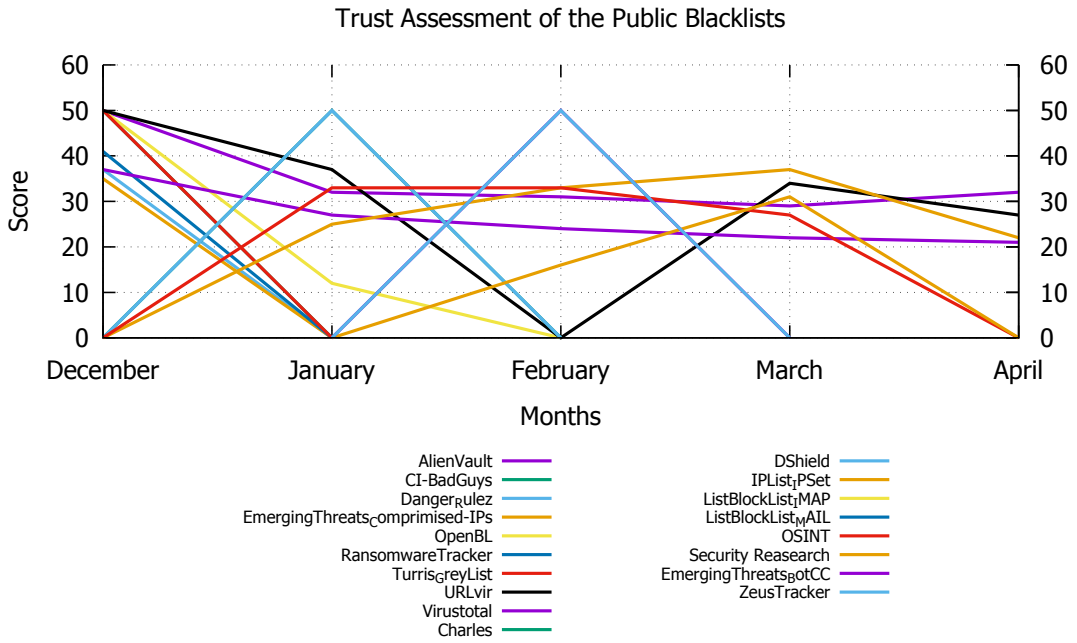


Figure 6.5: Trustworthiness Assessment of the blacklists over the five month period

In the end of this study about the blacklists, the blacklist that is more suitable for the EDP status and in the end had the highest trustworthiness is the Alienvault public blacklist.

We can conclude that the assessment of blacklists needs to be tuned. The assessment should have more consideration in a blacklist which has a presence in previous months

[1]Most of the blacklist had 100 of precision because they were related with a fewer number of cases, one or two, which were positive

that the new ones. We think that in our study that did not occur because the lists with previous presence were associated with more cases that the new ones, in average a new blacklist only had one or two cases related with, and had a lower precision that the new lists, therefore the assessment on the trustworthiness of the blacklists justifies the final results. If the weights of the history component have more into account the list with a historic from the ones without a history, the values between the appearances of new lists and the lists which already had cases in previous months could be different.

6.2.3 IP Addresses assessment

The IP Address assessment is the last study and the objective is to analyse the initial assessment, before the precision of the cases, and the final assessment considering the result of the cases of each IP address.

In order to verify if the Trustworthiness assessment module is correctly evaluating the IP and giving more weight to the IP addresses associated with the organization's positive cases, we additionally analyse the assessment components that influence the reputation of an IP address. The components analysed were: occurrence, term frequency, number of positives, number of false positives, IP address persistence over three months, and IP Position before and after it was associated with the incident cases. These components were already defined in Sect. 5.1.3. Tables 6.2 and 6.3 display all the values from these components between December 2016 and April 2017. Take into consideration that one case can have one or more target IP addresses, so the sum of all the TP and FP of each month in the Tab. 6.2 and Tab. 6.3 is higher than the actual total number of cases.

The first observation of the month of December is the increase of an IP's position when only is related with positive cases, one example are the IP addresses with an initial position 25 (lower than some other IP addresses) and due to their precision increase, after the assessment recalculation their position value increases to 75.

Other interesting aspect from the December's analysis is the difference of $IPPosition$ (5.5) between the IP 85.9.63.159 and the IP 119.81.124.89. Initially they have the same position, however after the assessment recalculation the position of the IP 85.9.63.159 is 75 and the position of the IP 119.81.124.89 is 73, two positions lower than the first IP address. This difference is due to the recalculation of the trust of the blacklists that reported those IP addresses. Although both IP addresses were confirmed as malicious by the blacklist Virustotal, the first IP was reported by the blacklist RansomwareTracker and the second was by Emergingthreats_compromisedips. As can be verified in Fig. 6.4a, the blacklist RansomwareTracker has a trust score of 41,67% and the Emergingthreats_compromised-ips has 35,71%. Both lists had five IP associated with true positive cases, the difference is in the number of IP associated with false positives cases. The blacklist Emergingthreats_compromised-ips had two against one of the Ransomware blacklist. This supports the $average$ component used in the assessment over the IP ad-

IP Address	Occ	TF	TP	FP	Presence in Month-3	Presence in Month-2	Presence in Month-1	Rank Before	Rank After
					December				
85.9.63.159	2	25,0%	1	0	false	false	false	25	75
185.84.65.226	2	25,0%	1	0	false	false	false	25	75
46.166.138.134	3	37,5%	1	0	false	false	false	38	79
119.81.124.89	2	25,0%	1	0	false	false	false	25	73
108.61.122.51	6	75,0%	1	0	false	false	false	75	100
104.200.151.80	2	25,0%	1	1	false	false	false	25	51
109.201.154.210	3	37,5%	1	1	false	false	false	38	56
66.96.149.1	2	25,0%	1	0	false	false	false	25	75
213.186.33.19	3	37,5%	1	0	false	false	false	38	82
213.186.33.24	2	25,0%	1	0	false	false	false	25	75
208.100.26.234	2	25,0%	0	1	false	false	false	25	29
					January				
160.153.129.210	2	16,6%	1	0	false	false	false	18	67
66.96.149.1	0	0,0%	1	0	false	false	true	32	67
74.220.199.6	2	16,6%	1	0	false	false	false	14	64
47.89.58.141	0	0,0%	1	0	false	false	true	32	67
81.169.145.88	2	16,6%	1	1	false	false	true	46	64
108.61.122.195	0	0,0%	1	0	false	false	true	32	67
179.43.176.66	1	8,3%	1	0	false	false	true	46	86
109.201.152.246	4	33,3%	1	0	false	false	true	57	100
104.238.169.143	0	0,0%	1	0	false	false	true	32	67
209.95.50.88	0	0,0%	1	0	false	false	true	32	67
52.4.209.250	3	25,0%	1	0	false	false	false	18	69
52.0.217.44	3	25,0%	1	0	false	false	true	51	91
213.186.33.19	0	0,0%	0	4	false	false	true	32	22
213.186.33.5	3	25,0%	0	1	false	false	true	53	46
66.175.58.9	0	0,0%	0	1	false	false	true	32	22
217.97.216.17	0	0,0%	0	1	false	false	true	32	22
213.186.33.40	0	0,0%	0	2	false	false	true	32	22
204.11.56.48	3	0,0%	0	1	false	false	false	18	24
91.199.120.14	3	0,0%	0	1	false	false	false	18	15
92.48.111.60	0	0,0%	0	1	false	false	true	32	22
81.169.145.162	2	16,6%	0	1	false	false	false	14	19
104.130.124.96	2	16,6%	0	1	false	false	false	14	14
134.0.11.63	0	0,0%	0	1	false	false	true	32	22
5.157.7.18	1	8,3%	0	1	false	false	true	46	41
46.166.188.244	3	25,0%	0	1	false	false	true	56	44
46.166.188.229	0	0,0%	0	1	false	false	true	32	22
46.166.190.183	0	0,0%	0	1	false	false	true	32	22
192.124.249.10	0	0,0%	0	1	false	false	true	32	22

Table 6.2: IP assessment over the months of the December (2016) and January (2017)

IP Address	Occ	TF	TP	FP	Presence in Month-3	Presence in Month-2	Presence in Month-1	Rank Before	Rank After
					February				
216.239.32.21	1	8,3%	2	0	false	true	false	34	81
69.172.201.153	5	41,7%	1	0	false	false	false	24	77
185.53.179.8	3	25,0%	0	1	false	true	false	43	42
23.227.38.32	1	8,3%	1	0	false	false	false	8	60
72.52.4.122	4	33,3%	1	0	false	false	false	21	67
198.252.100.188	0	0,0%	1	0	false	false	true	18	56
174.136.29.130	0	0,0%	1	0	false	false	true	18	56
198.185.159.144	0	0,0%	1	0	false	false	true	18	56
69.89.31.163	0	0,0%	1	0	false	false	true	18	56
213.186.33.19	0	0,0%	1	0	false	false	true	18	56
213.186.33.5	3	25,0%	1	0	false	true	true	64	100
213.186.33.40	0	0,0%	0	1	false	false	true	18	14
					March				
213.186.33.19	0	0,0%	1	0	false	false	true	8	47
66.96.149.32	1	9,1%	1	0	true	true	false	52	93
23.227.38.32	1	9,1%	1	1	true	false	false	35	59
185.53.179.8	2	18,2%	5	1	true	true	false	54	89
209.99.40.223	3	27,3%	1	0	true	true	false	59	99
64.29.151.221	0	0,0%	1	0	false	false	true	8	47
80.150.6.143	0	0,0%	1	0	false	false	true	8	47
216.239.32.21	1	9,1%	1	1	true	true	false	52	73
72.52.4.122	3	27,3%	0	1	true	false	false	43	44
187.45.240.41	0	0,0%	0	1	false	false	true	8	7
204.11.56.48	3	27,3%	0	1	true	true	false	62	58
198.185.159.144	0	0,0%	0	1	false	false	true	8	7
213.186.33.3	2	18,2%	1	1	true	true	true	66	80
46.105.57.169	1	9,1%	1	0	true	false	true	46	83
213.186.33.4	0	0,0%	0	1	false	false	true	8	7
81.169.145.161	0	0,0%	1	0	false	true	false	16	54
81.169.145.88	1	9,1%	1	0	true	true	true	60	100
66.96.149.1	0	0,0%	1	0	false	false	true	8	47
					April				
216.239.38.21	0	0,0%	3	0	true	true	true	49	97
213.186.33.19	0	0,0%	5	2	true	true	false	41	74
213.186.33.3	2	20,0%	1	1	true	true	true	65	94
216.239.34.21	0	0,0%	3	2	true	true	true	49	77
213.186.33.40	2	20,0%	2	0	false	false	false	17	71
216.239.32.21	0	0,0%	1	2	true	true	true	49	64
216.239.36.21	0	0,0%	1	2	true	true	true	49	72
213.186.33.5	3	30,0%	1	2	true	true	true	71	91
69.172.201.153	2	20,0%	0	1	true	true	false	57	58
64.29.151.221	0	0,0%	0	1	true	false	false	24	24
89.234.157.254	10	100,0%	0	0	true	true	true	100	100

Table 6.3: IP assessment over the months of the February, March and April of 2017

dresses (5.4).

January was the first month having IP persistence. Seven IP addresses were reported in December and January, 14 were reported only in December, and seven were only reported in January. Because of the persistence four alerts of malicious communications were true positive cases, thus improves the SOC capability in detecting infected assets. However one drawback was that having persistence also increased the number of false positives, 10 alerts were false positives. In February the persistence continued to have positive results, by having five IP addresses related with four true positive cases against one false positive case.

Being March the fourth month of our study, starting from this month, we will see the full three month history of the presence of an IP address and have more significant weight in the assessment. The results show that for the IP addresses that were not reported in the current month, but were reported in previous months, the final position do no go higher than 54,12. A cause for this can be because of the weights given for each month, as presented in Tab. 5.1.

April was the first month were IP addresses were discarded from the list. 63733 IP addresses were removed from the BADIP list, due to their inactivity. Taking into account this considerable number of removed IP addresses, the decrease of precision and the low number of IP addresses associated with cases, a question is raised about the period of preserving an IP address in the list. In further studies is necessary to evaluate if three months are enough to keep an IP address or if we extend the number of months to preserve the IP addresses the number of false positives cases increases.

Returning to the evaluation of the month of April, it was the first time, in our study period, that one IP address, address 89.234.157.254, with an initial position 100, i.e. the IP address with the highest reputation value, continues to be in position 100 after the recalculation of the assessment, and is dissociated with the cases. The reason for this is because the number of occurrences (Occ in Tab. 6.3) for the IP address is 10, the maximum number of occurrence of April, as for the IP addresses associated with cases the maximum number of occurrences is 3. The other factor is that the IP address appears in the three previous months having the maximum value of presence. This means that, although the organization did not had a case related with this IP address (or the communication between the organization and this IP was not detected), the IP should not be discarded and should be monitored with more attention because the initial assessment is that the IP 89.234.157.254 is the IP with the highest reputation value of the month of April, and after the recalculation of the assessment regarding the cases of April, this IP continues to be considered the IP with the highest reputation value of the month of April.

6.3 Prospective studies and discussion conclusions

As described in Sect. 5.2.2, we created a prototype SIEM rule to select the IP addresses with a position equal or higher than 85 from the BADIP list. The initial results show that this rule is giving a precision of 50%. Regarding the previous study one dilemma appears in the selection of the rule over the BADIP list: IP addresses with a low trust's value were related with true positive cases. So in order not to discard theses IP addresses (and cases) is necessary to rethink and provide weights to the Trustworthiness assessment's components. We think that by having weights in some components the IP addresses that are reported by the blacklist in the current month will have a higher value and will not be discarded. To tune the rule to capture all these positive cases the Trustworthiness assessment module must also be tuned based on the results of this study. We did not present here the study of the rule because is in its initial phase and we need more results for a study.

We can conclude that this study provides valuable information about the Trustworthiness assessment module, the blacklist and the BADIP list. We find that by using new components which consider the incidents of the organization related with suspicious communications, our list had a higher precision that the OSINT-LIST, a list that gathers information from public and private blacklist, and have IP addresses that the SOC considers as malicious. The BADIP list precision value is closer to the ArcSight Global Cases precision value. The BADIP list average precision is only 1,24% lower than the ArcSight Global list, which contains paid cybersecurity appliances, private lists and the OSINT-LIST. We also observe that the Alienvault, Virustotal and the IP_List_IPSet are the blacklists more appropriate for the organization reality. We observe that the components of the Trustworthiness assessment should be tuned, is necessary to give weights to each component and test these weights results in future studies in order to improve the precision of the BADIP list.

Other aspect which requires investigation is the decision of a case being true or false positive. In most months we have IP addresses related with TP and FP cases. In the same month, for a period of time, one IP address can be malicious and for other period of time the same IP address can be not malicious. Justifying the reason why an IP address can be associated with TP and FP cases in the same month. However, this is only one hypotheses. To validate this, or to reinforce the decision of classifying a case as true or false positive, more detection methods are required to do a full scan and analysis on the organization assets to verify if the asset is infected or, was infected and the antivirus removed the infection, or even in some moment the initial infection was blocked by some cybersecurity appliance. This topic could be a future development to improve the capabilities of the SOC in detection and assessment of a case, using our approach.

Chapter 7

Conclusion & Future Work

The main goal of this work was to improve the SOC daily operations, by enhancing the system used to monitor and alert security events, the SIEM system.

We developed a Threat Intelligence solution to be used with a SIEM system and to improve its alarmist capabilities. Our TI uses Security Metrics and a Trustworthy Blacklist framework to achieve that goal. With our Threat Intelligence solution, the Security Operations team can obtain information about the security status in the organization and, trustworthy and meaningful information about malicious IP addresses, allowing them to be more effective.

We have established a set of security metrics, with a defined purpose and with a taxonomy structured, to be applied with a SIEM system which we think that every information security team must have to augment the knowledge about the state of cybersecurity. In addition to the SM we created two visualization prototypes that exhibit comparable information between the current and the previous month.

One security metric that we think that will improve the awareness of the vulnerabilities and the dependencies between the assets is the surface metric. This metric calculates an overall risk of the organization concerning a known cyberthreat and provides information about a spread of a cyberattack in the organization, regarding the dependencies between the assets and their vulnerabilities. Although, this metric is not calculated automatically and there is no work about the implementation of the metric in a SIEM system, we think that the metric can use our TI solution to gather external information about current cyberthreats and from the SM the assets vulnerable to them and the dependencies between the assets.

The Trustworthy Blacklists is an innovation when assessing the blacklists and their information. In addition to assessing the IP addresses with information provided by all the blacklists, a component of evaluation already used by other frameworks described in the related work, our framework also considers the inside information from the SOC operation, detected security incidents and vulnerabilities, i.e. true or false positive, with a history about the blacklists and the IP addresses.

The Trustworthiness framework has been tuned and evaluated over a five month period. We have 121 public blacklist from which we gather information every day, and each month we have, in average, 166750 IP addresses to classify their trustworthiness of maliciousness.

We did a practical study experience at a worldwide company such is the EDP, to evaluate the performance and to analyse the results of our framework. The practical experience consisted in: 1) select the incident cases related with the suspicious communications from two lists: the OSINT-LIST, which is a list with public-private information, and the ArcSight Global list, which is a list with all the suspicious communications incidents cases detected by paid cybersecurity tools and the OSINT-LIST; 2) assess those selected incidents; 3) compare the results with our list.

The results demonstrated that, over time, the public blacklists started to have a trust score, and with that the framework started to prioritize IP addresses reported by blacklists with a higher score. This is a good result because since the beginning we wanted to classify the trustworthiness of public blacklists not for their online reputation but for their efficiency and results within the organization.

Over the course of our study and the analysis of the SOC operation incidents, we determine that the persistence component, which covers the persistence of an IP address being reported by the blacklists or its precision is greater than zero, would increase the number of positive cases. Our analysis confirms this, and by only using the persistence component, our framework increases the detection of communications that the company OSINT-LIST didn't caught.

We concluded our analysis by comparing the precision of our assessed list with the company OSINT-LIST and the ArcSight Global. The results support our initial statement that organizations should have a component to assess the information gathered from their public-private lists considering the organizations status, thus improving the detection's precision over suspicious communications.

We use public (free) blacklists where we did an assessment and compare its precision with a list which has public and private blacklists, but without the organization assessment, and the results shown that with the assessment we can have a higher precision that the list used by the organization. The difference between the two precisions is almost 3 percent.

Over the paid cybersecurity tools used by the SOC team, our list had approximately the same precision, with a difference of less than 1,3% lower of the list with the paid cybersecurity tools.

The $persistence$ and the $precision$ metrics are the main factors for the BADIP and BADIPPotential good precision, due to its consideration of the internal information of the incidents operations by the SOC team to classify, maintain or discard an IP address, increased our list coverage and reduced the false positives.

We can also conclude that the blacklist with the highest trust regarding the cases occurred between December of 2016 and April of 2017 is from AlienVault.

We think that if our framework was operational and used by the SOC team, some of the repeated false positive cases would not occur due to the reassessment of the IP addresses. The reassessment would decrease the IP address reputation, thus its position would also decrease, and, therefore, more communications would be required for the SIEM rule to trigger the alarm about a suspicious communication. However, we need to rethink about the assessment process and how the rules select the IP addresses. In our study, there were situations where an IP address had an initial low score but was related with true positive cases of the organization. These IP had a low position, because they did not have history values, so their reputation value was low comparatively with the IP addresses that had history. Because of their low reputation the SIEM rule could take more time, requiring more communications, to trigger the alert, increasing the Time To Detect (TTD) of the SOC's team.

With these conclusions, the next steps to improve the Trustworthiness Blacklists framework is to tune the current parameters in the assessment, each of them with weights of importance. This can enhance the initial evaluation to be more accurate to the reality. We can also adapt the IP Collector to categorize the IP addresses for their type of maliciousness (e.g. C&C, ransomware, phishing, and more). Each type of categorization has a weight, depending on the organization's exposure to the threat. We do not want to just go to IP addresses, our program can assess other types of information that the blacklists provide, such as domains, emails or URL. We also can include more than public blacklists, i.e. Twitter, Facebook or other social engines. To accomplish this is necessary to adapt the IP Collector to collect these types of information and use the concept of SOCMINT [32]. The DiSIEM project has some parallel works, under development, that cover this concept.

The SIEM rule module can also be improved. The SIEM rules can select the IP addresses by their level of reputation trust, and by their type of maliciousness. With these detailed rules the information security team can understand what type of threats the organization is more likely to have and react with the right measures to mitigate and clean the infections by threat. This knowledge can help reducing the number of false positives. In addition to gathering the normal information about the IP and the type of threat that the IP is associated, the IP's collector can be modified to collect intelligence about that threat, i.e. the ports usually used to communicate. If the IP Collector adds this information to the BADIP file, the rules could use it to filter the security events which this behaviour does not appear, hence reducing the number of false alerts.

Another step on improving the SIEM rules is the forecasting of the rate of false positives, through increasing or reducing the threshold used by the rules to select the IP by their level of reputation. Each rule has definitions about the number of required commu-

nications between the organization and one or more malicious IP address of the rule, in an interval of time (more detailed information in Section 2.3.1). In future work a program could be developed to estimate the number of possible false positives if the threshold of the SIEM rule is increased or decreased. If the rate of false positives was reduced, the program would inform the information security team about the augment of performance by changing the settings of the rule or in an autonomous way the program could sets the new rule's definitions.

In conclusion, the objectives of the work were successfully accomplished. Further, we already have new ideas for future work that combined with the preliminary studies, can help to improve the results obtained. There are several options and areas that can be improved, and each module we present can be implemented in our framework. We expect this to improve Threat Intelligence and, consequently, strengthen the capabilities of the SIEM, leading to enhanced security at the organization.

References

[1] (2017). SOC Capabilities. Retrieved July 6, 2017, from https://pbs.twimg.com/media/C91IB9wXUAAxraR.jpg.

[2] AlienVault (2016). *AlienVault Open Threat Exchange (OTX) ™ User Guide*. AlienVault.

[3] ArcSight (2010). WhitePaper: Security Operations Metrics Definitions for Management and Operations Teams. Technical report, ArcSight.

[4] Berinato, S. (2005). A few good information security metrics. Retrieved October 24, 2016, from http://www.csoonline.com/article/2118152/metrics-budgets/a-few-good-information-security-metrics.html.

[5] Bromiley, M. (2016). Threat Intelligence: What It Is, and How to Use It Effectively. *SANS Institute Reading Room site*.

[6] Butler, J. M. (2009). Benchmarking security information event management (SIEM). *A SANS Whitepaper*.

[7] Cain, C. I. and Couture, E. (2011). Establishing a Security Metrics Program. Technical report, GIAC Enterprises.

[8] Chuvakin, A. (2014). On SIEM Tool and Operation Metrics. Retrieved October 28, 2016, from http://blogs.gartner.com/anton-chuvakin/2014/06/17/on-siem-tool-and-operation-metrics/.

[9] CIS (2010). The CIS Security Metrics - v1.1.0.

[10] Cornell, C. (2015). Five Metrics You Should Be Recording for Incident Response. Retrieved October 28, 2016, from https://swimlane.com/five-metrics-for-incident-response/.

[11] CYBERGYM (2016). *Arcsight user guide trainees*. CYBERGYM.

[12] DiSIEM (2017). Disiem project. Retrieved July 12, 2017, from http://disiem-project.eu/.

[13] EDP (2009). EDP - História da Marca. Retrieved October 28, 2016, from https://www.edp.pt/pt/aedp/sobreaedp/marcaEDP/Pages/HistoriaMarca.aspx.

[14] EDP (2016). EDP - Eólicas. Retrieved October 28, 2016, from http://www.edp.pt/pt/aedp/unidadesdenegocio/energiasrenovaveis/Pages/EnergiasRenovaveis.aspx.

[15] EllisLab (2017). COdeIgniter. Retrieved July 10, 2017, from https://www.codeigniter.com/.

[16] Enisa (2017). Incident handling automation. Retrieved June 27, 2017, from https://www.enisa.europa.eu/topics/csirt-cert-services/community-projects/incident-handling-automation.

[17] Forbes (2016). The world's Largest Companies 2016. Retrieved October 28, 2016, from http://www.forbes.com/global2000/.

[18] Foundation, M. (2017). Mariadb. Retrieved June 14, 2017, from https://mariadb.org/.

[19] Gordon, S. (2015). Operationalizing Information Security Putting the top 10 SIEM best Practices to Work - Process, Metrics and Technology Considerations.

[20] HPE (2015). *ArcSight - Logger Administrator Guide*. Hewlett Packard Enterprise (HPE).

[21] HPE (2016). Software Solution Designed to Scale — HPE Software. Retrieved October 28, 2016, from https://saas.hpe.com/en-us/home.

[22] INForum (2010). Apresentação INForum 2017. Retrieved August 9, 2017, from http://inforum.org.pt/INForum2017.

[23] Jansen, W. (2009). Directions in security metrics research. Technical report, National Institute of Standards and Technology.

[24] Jaquith, A. (2007). *Security Metrics: Replacing Fear, Uncertainty, and Doubt*, volume 1. Addison-Wesley.

[25] Johnson, L. K. (2007). *Handbook of intelligence studies*. Routledge.

[26] Julisch, K. A unifying theory of security metrics with applications. *IBM Research–Zurich*, pages 1–19.

[27] Kaur, M. and Jones, A. (2008). Security Metrics - A Critical Analysis of Current Methods. In *Proceedings of the 9th Australian Information Warfare and Security Conference, Edith Cowan University, Perth Western*.

[28] Kotenko, I. and Novikova, E. (2014). Visualization of security metrics for cyber situation awareness. In *Availability, Reliability and Security (ARES), 2014 Ninth International Conference on*, pages 506–513. IEEE.

[29] Kotenko, I., Polubelova, O., Saenko, I., and Doynikova, E. (2013). The ontology of metrics for security evaluation and decision support in siem systems. In *Availability, Reliability and Security (ARES), 2013 Eighth International Conference on*, pages 638–645. IEEE.

[30] Kührer, M., Rossow, C., and Holz, T. (2014). Paint it black: Evaluating the effectiveness of malware blacklists. In *International Workshop on Recent Advances in Intrusion Detection*, pages 1–21. Springer.

[31] Muthukrishnan, S. M. and Palaniappan, S. (2016). Security metrics maturity model for operational security. In *Computer Applications & Industrial Electronics (ISCAIE), 2016 IEEE Symposium on*, pages 101–106. IEEE.

[32] Omand, D., Bartlett, J., and Miller, C. (2012). Introducing social media intelligence (SOCMINT). *Intelligence and National Security*, 27(6):801–823.

[33] Payne, S. C. (2006). A guide to security metrics. *SANS Institute Reading Room site*.

[34] Rathbun, D. (2009). Gathering Security Metrics and Reaping the Rewards. *SANS Institute Reading Room site*.

[35] Rossow, C., Czerwinski, T., Dietrich, C. J., and Pohlmann, N. (2010). Detecting gray in black and white. MIT Spam Conference.

[36] Savola, R. M. (2007). Towards a taxonomy for information security metrics. In *Proceedings of the 2007 ACM workshop on Quality of protection*, pages 28–30. ACM.

[37] Sinha, S., Bailey, M., and Jahanian, F. (2008). Shades of grey: On the effectiveness of reputation-based "blacklists". In *Malicious and Unwanted Software, 2008. MALWARE 2008. 3rd International Conference on*, pages 57–64. IEEE.

[38] Slayton, R. (2015). Measuring risk: Computer security metrics, automation, and learning. *IEEE Annals of the History of Computing*, 37(2):32–45.

[39] Tashi, I. and Ghernaouti-Hélie, S. (2008). Efficient security measurements and metrics for risk assessment. In *Internet Monitoring and Protection, 2008. ICIMP'08. The Third International Conference on*, pages 131–138. IEEE.

[40] Tzu, S. (2005). *The art of war*. Shambhala Publications.

[41] Vaarandi, R. and Pihelgas, M. (2014). Using security logs for collecting and reporting technical security metrics. In *Military Communications Conference (MILCOM), 2014 IEEE*, pages 294–299. IEEE.

[42] Vaughn, R. B., Henning, R., and Siraj, A. (2003). Information assurance measures and metrics-state of practice and proposed taxonomy. In *System Sciences, 2003. Proceedings of the 36th Annual Hawaii International Conference on*, pages 10–pp. IEEE.

Appendix A

DiSIEM - SM survey

Security Metrics – exposition and survey

1 Introduction on Security Metrics – exposition and survey

The purpose of this document is to define, in the scope of Security Metrics (SM), which metrics each partner considers relevant, the corresponding input data that can be provided and how it is generated. The survey provides information about the selected SM, which are in the context of the project and will be useful for organizations with SIEM systems. In this survey a total of 63 metrics were selected, from which two were originally created by us 4 additional variations were also created for the two metrics created.

2 Metrics

We can divide the SOC capabilities into three main sectors: People/Management where we evaluate the SOC team work, training, their response to incidents and structure, and the management process. Process, where we monitor the incidents, vulnerabilities cases, incident analysis and resolution. Technologies, where we analyse the network infrastructure, the vulnerability track, the SIEM infrastructure and the log management. Security metric must be in the same perspective and direction. We structure all the gathered metrics into these three main topics.

2.1 People/Management

2.1.1 UA - User Activity [1]

Definition: The UA metric calculates the top users (usually the top 10) with biggest number of failed logins attempts. This metric helps to detect (patterns of) malicious activity.
Input data: The events which contain the users and their failed logins attempts.
Output: A list containing ten users with biggest number of failed attempts and the corresponding number.
Suggested frequency: Daily

2.1.2 PUA - Privileged Users Activity [1][6]

Definition: The PUA metric computes the top privileged users who have the biggest number of logins. This metric helps the SOC team to detect abnormal activity from these users.
Input data: The logs containing the successful login attempts from privileged users.

Output: A list of the top 10 privileged users who have the biggest number of logins and the corresponding numbers.
Suggested frequency: Daily.

2.1.3 PETVI(a) - SOC's Percentage of effort Time to resolve Vulnerabilities and resolve Incidents

Definition: The PETVI metric calculates in percentage, the SOC's team effort time to resolve the vulnerabilities. This metric can be used with the Efficacy metric (2.2.12) to obtain a view of the SOC's team performance.
Input data: all the SOC's team work time and the team's time deliverable to resolve the vulnerabilities.
Output: percentage of the team's effort time to resolve the vulnerabilities and resolve incidents.
Suggested frequency: Monthly

2.1.3.1 PETmV - SOC's Percentage of effort time to resolve Vulnerabilities

Definition: The PETmV metric is a sub metric of the PETVI(a)(2.1.3), and only focus on calculating the percentage of effort time in resolving the vulnerabilities.
Input data: total SOC's team work time and the time to resolve the vulnerabilities.
Output: percentage of the team's effort time to resolve the vulnerabilities.
Suggested frequency: Monthly.

2.1.3.2 PETrI – SOC's Percentage of effort time to resolve incidents

Definition: The PETrI metric is a sub metric of the PETVI(a) (2.1.3), and only focus on the percentage of effort time on resolving incidents.
Input data: total SOC's team work time and the time of to resolve the incidents.
Output: Percentage of the team's effort time to resolve the incidents.
Suggested frequency: Monthly.

2.1.4 PETVI(b) - SOC's Percentage of effort Time to resolve Vulnerabilities and resolve Incidents

Definition: This metric is similar to the PETVI(a) (2.1.3). Contrary with the PETVI(a) (2.1.3) which takes into consideration the vulnerabilities and incidents prior created and resolved this month. The PETVI(b) metric only calculates the effort for the vulnerabilities resolved and incidents resolved which were created and resolved this month.
Input data: all the team's work time and the time to resolve the vulnerabilities and incidents of that period (month).
Output data: the team's effort time to resolve vulnerabilities and resolve incidents of that period (month).
Suggested frequency: Monthly.

2.1.5 CU - Cost of Updates [1]

Definition: The CU metric calculates the cost of an update. The number (and required time) of signature, policy, application and other software updates. This metric helps the security manager to explain the effort and amount of time involved in updating the various security devices and agents.
Input data: Number of software updates and average length time for each update over the time and unitary cost of work and update.

Output: Total cost of updates.
Suggested frequency: Monthly.

2.1.6 AC - Asset Criticality [7]

Definition: The AC metric calculates a value to represent the impact for the organization resulting from the loss of an asset. The value can be quantifiable or qualitative (Low, Medium, High). As assets we consider: hosts (middleware, firewalls, IPS, IDS, databases) and applications.
Input data: a list of the assets and information about them (their supply for the company, value for the company, their dependencies, etc.).
Output: a value for the criticality of the asset.
Suggested frequency: Monthly or when there are changes in the list of assets.

2.1.7 BV - Business Value [7]

Definition: The BV metric calculates a value to represent the impact from the loss of a business or a service to the organization. The metric can be quantifiable or qualitative (Low, Medium, High).
Input data: a list of businesses or services and information about them (their value for the company, their dependencies from other business or services, applications, etc.).
Output: A value representing the impact from the loss associated to a business or service.
Suggested frequency: Monthly or when changes in business/services occur.

2.1.8 RRSO - Rate of return for security operations (derived from [1])

Definition: The purpose of this metric is to show the importance of investment in the security operations, by comparing the overall cost of security operations to losses due to security incidents. It is the percentage ratio between the total costs with incidents resolution and the costs of security operations.
Input data: The total number of incidents resolved, the average cost per incident and total costs with security operations.
Output: The percentage ratio between the total costs with incidents resolution and the costs of security operations.
Suggested frequency: Monthly.

2.2 Processes

2.2.1 MTTR - Mean Time to Remediate (a known vulnerability and a reported incident) (derived from [7])

Definition: MTTR is the average time the team spent to resolve a 'problem' (here the 'problem' is the disjunction of known vulnerabilities and reported incidents). Allows to assess the efficiency of resolution vulnerabilities/incidents and provides the manager with quantifiable information to request, if necessary, more personal or equipment to improve (reduce) the MTTR.
Input data: date of discovery of every known vulnerability/incident and their date of the resolution of these vulnerabilities and incidents.
Output: average number of days (can be other type of time measurement) that the team spends to resolve the 'problem'.
Suggested frequency: Daily.

2.2.1.1 *MTTRV(a) - Mean time to resolve a vulnerability*

Definition: MTTRV is a particular case of MTTR and consists in measuring the average time which a known vulnerability is resolved.
Input data: date of discovery of every known vulnerability and their date of resolution.
Output: average number of days (can be other type of time measurement) that the team spends to resolve a known vulnerability.
Suggested frequency: Daily.

2.2.1.2 *MTTRI(a) - Mean time to resolve an incident*

Definition: MTTRI is another particular case of MTTR and consists in measuring the average time for the resolution of incidents.
Input data: dates of the reported incidents and their resolution dates.
Output: average number of days (can be other type of time measurement) that the organization spends to resolve an incident.
Suggested frequency: Daily.

2.2.2 Age of the oldest known vulnerability and not resolved by severity

Definition: This metric consists in computing the age of the oldest unresolved known vulnerability by each category of vulnerability severity. This helps the team to assess and take priority action in the vulnerabilities by their age and severity category.
Input data: dates of all vulnerabilities which are known and not resolved.
Output: number of days (can be also minutes, hours, months, etc..) of the oldest unresolved vulnerability for each severity type.
Suggested frequency: Daily.

2.2.3 Number of known vulnerabilities and not resolved by severity

Definition: This metrics consists in measuring the status of existing vulnerabilities by their severity levels, counting the total number of known unresolved vulnerabilities for each severity category.
Input data: dates of all the known unresolved vulnerabilities for each severity category.
Output: number of unresolved vulnerabilities by their severity level.
Suggested frequency: Daily.

2.2.4 Number of known unresolved vulnerabilities by vulnerability type

Definition: The metric measures the total number of known unresolved vulnerabilities for each vulnerability type. With these values, the team can manage their efforts and resources to resolve vulnerabilities by knowing how many vulnerabilities are open, by vulnerability type.
Input data: dates of open vulnerabilities for each vulnerability type.
Output: total number of known unresolved vulnerabilities by vulnerability type.
Suggested frequency: Daily.

2.2.5 Number of vulnerabilities cases by month in each severity category

Definition: This metric lets the team to know and report, for each month the number of vulnerabilities identified by each severity category.
Input data: number of vulnerabilities cases identified for the current month for each severity category.

Output: total number of vulnerability cases identified for the current month for each severity category.
Suggested frequency: Monthly.

2.2.6 Number of vulnerabilities cases by responsible

Definition: This metric measures the total number of vulnerabilities cases each responsible (owner) of the asset/application has. It can be change to only provide the number of vulnerabilities cases which are still to be resolved.
Input data: vulnerabilities cases of each responsible.
Output: total number of vulnerabilities cases of each responsible.
Suggested frequency: Monthly.

2.2.7 Number of assets tested by month

Definition: This metric computes the number of assets which were tested to verify if they had vulnerabilities.
Input data: assets tested in the respective month.
Output: total number of assets tested for that month.
Suggested frequency: Monthly.

2.2.8 Number of vulnerabilities identified by tested asset

Definition: This metric computes the number of vulnerabilities identified for each tested asset. This metric can be changed to the number of vulnerabilities identified in each tested asset, by their severity category. A correlation of a set of results of this metric will show the most vulnerable tested assets.
Input data: vulnerabilities identified for each tested asset.
Output: total number of vulnerabilities for each tested asset.
Suggested frequency: Monthly.

2.2.9 Number of vulnerabilities identified and reported incidents, by month

Definition: This metric calculates the total number of 'problems' (vulnerabilities and incidents). This metric counts the total number of cases, discarding if where already resolved or are still to be resolved. The metric can be changed to count the total number of vulnerabilities identified and reported incidents which are not yet resolved or are already resolved. These two-additional metrics can be change to provide the results for each month or for the global scenario.
Input data: all the cases of the month.
Output: total number of cases, opened and closed, of the month.
Suggested frequency: Monthly.

2.2.9.1 Number of reported incidents by month

Definition: This metric is a particular case of the previous one. It counts the total number of reported incidents for each month. This metric can be changed to just count the number of reported incidents which are still to be resolved or those which are already resolved. The metric can also generate sub-metrics for each type of incident (phishing, malicious attack, unauthorized access, etc.).
Input data: the reported incidents of the month.
Output: total number of reported incidents of the month.
Suggested frequency: Monthly.

2.2.10 Number of reported incidents at each region of operation

Definition: This metric computes the number of reported incidents for each region of operation.
Input data: reported incidents.
Output: total number of incidents reported by each region of operation.
Suggested frequency: Monthly.

2.2.11 Number of resolved incidents and vulnerabilities by month

Definition: This metric calculates the number of resolved incidents and vulnerabilities resolved by the SOC team and other teams.
Input data: the cases resolved/closed (true positive) incidents and vulnerabilities in that month.
Output: total number of cases resolved.
Suggested frequency: Monthly.

2.2.12 Efficacy of resolution of incidents and vulnerabilities(a)

Definition: This metric calculates the efficacy, in that month, of the SOC team and other teams involved in resolving incidents and vulnerabilities. The result should be kept in history to be compared for the following months, thus allowing observing and correlating the line of effort of the team.
Input data: all the cases opened until that month and the cases closed in that month.
Output: the ratio between the total cases resolved and the total opened cases for that period.

$$E_f = \frac{R_C}{Total_C},$$

Where:
E_f – Efficacy of the team resolving cases
R_C – Resolved cases in that period
$Total_C$ – Total cases in that period (resolved cases on that month + open cases that month)
Suggested frequency: Monthly.

2.2.12.1 Efficacy of resolution of vulnerabilities

Definition: This metric calculates the efficacy, in that month, by the SOC team and other teams involved in resolving vulnerabilities. The result should be kept in history to be compared for the following months, thus allowing observing and correlating the line of effort of the team. Contrary to the previous metric (2.2.12) this metric only concerns in calculating the teams' efficacy for vulnerabilities resolution.
Input data: all the vulnerabilities of that period.
Output: the ratio between the total number of vulnerabilities resolved and the total number of opened vulnerabilities for that period.

$$E_f = \frac{R_C}{Total_C},$$

Where:
E_f – Efficacy of the team in vulnerabilities resolution
R_C – Resolved vulnerabilities in that period

$Total_C$ – Total number of vulnerabilities in that period (Resolved vulnerabilities + Opened vulnerabilities)
Suggested frequency: Monthly.

2.2.12.2 Efficacy of resolution of incidents

Definition: This metric calculates the efficacy, in that month, by the SOC team and other teams involved in resolving incidents. The result should be kept in history to be compared for the following months, thus allowing observing and correlating the line of effort of the team. Contrary to the metric (2.2.12) this metric only concerns with calculating the teams' efficacy for incidents resolution.
Input data: all the incidents of that period.
Output: the ratio between the total number of incidents resolved and the total number of incidents opened in that period.

$$E_f = \frac{R_C}{Total_C},$$

Where:
E_f – Efficacy of the team resolving incidents
R_C – Resolved incidents in that period
$Total_C$ – Total incidents in that period (resolved incidents + incidents cases)
Suggested frequency: Monthly.

2.2.13 Efficacy of resolution of incidents and vulnerabilities(b)

Definition: This metric considers incidents and vulnerabilities, which were opened and closed in that month, and calculates the efficacy of the SOC team and other teams involved in their resolution. The result should be kept in history to be compared for the following months, thus allowing observing and correlating the line of effort of the team. To observe and correlate the line of effort of the team.
Input data: all the cases of that period.
Output: the ratio between the total cases resolved and the total opened cases for that period.

$$E_f = \frac{R_C}{Total_C},$$

Where:
E_f – Efficacy of the team
R_C – Resolved cases in that period
$Total_C$ – Total cases in that period (resolved cases + opened cases)
Suggested frequency: Monthly.

2.2.14 PIS - Percentage of infected Systems (derived from [1][7])

Definition: The PIS metric tracks the occurrences of systems (or assets) infected by malware or with vulnerabilities. It calculates the percentage of infected systems, by different malware infection or independent vulnerabilities, in the organization.
Input data: systems' name and their security status (infected or clean).
Output: Percentage of the infected systems, by malware infection or vulnerabilities type.
Suggested frequency: Monthly.

2.2.15 TMA - Top malware activity [1][6]

Definition: The TMA computes the top malware detected in the organization by their criticality.
Input data: Reported incidents with malware activity.
Output: Top (ex.: top five or top ten) malware activities and their criticality that were detected inside the organization.
Suggested frequency: Daily.

2.2.16 Attacks classified by their criticality (derived from [6])

Definition: This metric calculates the number of attacks made, by their criticality, against the vulnerable systems. Provides a view about the attackers and how vulnerable the organization is.
Input data: Reports of incidents and/or events.
Output: Number of attacks made, by their criticality, against vulnerable systems.
Suggested frequency: Monthly.

2.2.17 TEE - Top Egress Event [1]

Definition: The TEE metric considers the SIEM events and calculates the top ten source IPs, destination IPs and destination ports for events leaving the organization with malicious activity, originating from within the organization. This metric helps the SOC team to analyse and identify patterns of malicious activity originating from the organization.
Input data: communication events leaving the organization, provided by the SIEM, containing the source IPs, destination IPs and destination ports.
Output: a list of the top 10 source IPs, destination IPs and destination ports leaving the organization
Suggested frequency: Daily.

2.2.18 TIE - Top Ingress Event [1]

Definition: The TIE metric is similar to the TEE metric and also uses the SIEM events. It calculates the top 10 source IPs, destination IPs and destination ports with malicious intent. It focuses in the communications which the source is the internet and the destination is the organization. This metric helps the SOC team to analyse and identify patterns from malicious activity.
Input data: events of communication from the internet to the organization, provided by the SIEM, and containing the source IP, destination IP and destination port.
Output: a list of the top 10 communication events from the internet to the organization.
Suggested frequency: Daily.

2.2.19 TFA - Top Foreign attacks [1]

Definition: The TFA calculates the top 10 most severe attacks originating from foreign countries.
Input data: security events (attacks) that lead to an incident, their severity levels and their origins.
Output: a list of the top 10 most severe attacks.
Suggested frequency: Daily.

2.2.20 TFC - Top Foreign Countries [1]

Definition: The TFC metric calculates the top 10 countries destinations with communication from the organization and the top 10 countries sources with traffic incoming to the organization.
Input data: events containing the source and destination country.
Output: the top ten destination countries and the top ten source countries.
Suggested frequency: Daily.

2.2.21 AS - Attack Surface [7]

Definition: The AS metric calculates the potentiality of occurrence of an attack using the system resources and their interdependencies. These resources can be critical or not critical entry/exit points, channels, vulnerable subsets/applications of the system, untrusted data items sent, etc. The risk of the system is directly connected with the attack surface, hence if the attack surface increases the risk will also increase. Each resource contributes for the calculation of the attack surface value by their Damage Potential-Effort Ratio [7].
Input data: name of the resources, the dependencies between them and the risk associated.
Output: value of the attack surface can be two things. A risk's value of the system (considering he risk of the sub-systems), it can be a quantitative or qualitative value. Percentage of the system infected by an possible attack, concerning the system and sub-systems vulnerabilities and dependencies, and the effort and damage of an attack.
Suggested frequency: Monthly.

2.2.22 FE - Firewall Entry [1]

Definition: The FE metric calculates the top external blocked sources which exceeded the reasonable number of blocked sessions permitted.
Input data: the blocked IPs.
Output: the top 10 external sources by block/permitted ratios.
Suggested frequency: Daily.

2.2.23 TAFD - Top access failures by destination [6]

Definition: This metric calculates the top ten destination access failures.
Input: A list of events.
Output data: The top ten access failures by destination (IP address or hostname).
Suggested frequency: Daily.

2.2.24 TAFBU - Top access failures by business unit [6]

Definition: This metric focus in determining the top ten access failures by business unit.
Input data: A list of events.
Output data: Top ten access failures by business unit.
Suggested frequency: Daily.

2.2.25 IUH - Installation of unauthorized hardware (derived from [3])

Definition: The IUH metric calculates three factors related with unauthorized hardware/device: the average number of hours an unauthorized hardware/device is

plugin into the network, the total number of unauthorized hardware/devices connected in the organization's network and, lastly unauthorized hardware/devices threat level. To calculate the threat level a set of steps are required. The first is a list of unauthorized hardware/devices and their STL - Security Threat Level – being the one the smallest severe threat and the five the most important severe threat. Then a device discovery scan is made to identify the unauthorized devices and when they were installed. After this process the calculation for the threat level begins. The devices with the same STL are grouped. For each group the STL is multiplied by the number of unauthorized devices (group's length). Then it's summed the average plugin hours of the group. The risk score is calculated by the sum of each group threat level multiplied by the equalizer controller threat level (TL). This TL is a score, chosen by the organization, for the threat level by having unauthorized devices connected in the network. This control threat level should be between one to ten

$$[(STL_{Level\ 1} \times D_{UNAUTH}) + AHN] + [(STL_{Level\ 2} \times D_{UNAUTH}) + AHN] + [(STL_{Level\ 3} \times D_{UNAUTH}) + AHN] * TL = \text{Devices Threat Level (Risk Score)}$$

where
D_{UNAUTH} = Number of unauthorized devices discovered in a given period
STL = Security Threat Level, on a scale from 1-5, 5 being a high importance (consider the device threat/risk for the organization.
AHN = Average Hours on Network
TL – Threat Level

Input data: The unauthorized devices types and their STL for the organization, the result of the device discovery scan.
Output: Total number of unauthorized devices, average number of hours an unauthorized device is plugin and the unauthorized device threat level
Suggested frequency: Monthly

2.2.26 IUS - Installation of unauthorized software (derived from [3])

Definition: Similar to the IUH, it calculates the three factors related with unauthorized software. The average number of hours an unauthorized software installed, the total number of software installed, lastly unauthorized software threat level. To calculate the threat level a set of steps are required. The first is a list of unauthorized software and their STL - security threat level – being one the smallest severe threat and five the most important severe threat. Then a software discovery scan is made to identify the unauthorized software and determine when they were installed. After this process the calculation for the threat level begins. The devices with the same STL are grouped. For each group the STL is multiplied by the number of unauthorized devices (group's length). Then it's summed the average plugin hours of the group. The risk score is calculated by the sum of each group threat level multiplied by the equalizer controller threat level (TL). This TL is a score, chosen by the organization, for the threat level by having unauthorized devices connected in the network. This control threat level should be between one to ten.

The software with the same STL are grouped and for each group is multiplied the number of unauthorized software of each group (group's length) and then is

summed the average hours in the network of the group. The results of each group are summed up together. The final sum is then multiplied by an equalizer number, for the formula to be in the interval of [1,10], being ten the most severe threat level.

$$[(STL_{Level\ 1} \times D_{UNAUTH}) + AHN] + [(STL_{Level\ 2} \times D_{UNAUTH}) + AHN] + [(STL_{Level\ 3} \times D_{UNAUTH}) + AHN] * TL = \text{Devices Threat Level (Risk Score)}$$

where
D_{UNAUTH} = Number of unauthorized software in a given period
STL = Security Threat Level, on a scale from 1-5, 5 being a high importance (consider the device threat/risk for the organization.
AHN = Average Hours on Network
TL = Threat Level

Input data: The unauthorized software types and their STL for the organization, the result of the software discovery scan.
Output: Total number of unauthorized software, average number of hours an unauthorized software is installed and the unauthorized software threat level
Suggested frequency: Monthly

2.2.27 SHS - Security "Health" Score [1]

Definition: The SHS metric computes a weighted sum of several statistics regarding antivirus statistics and logs, ingress and egress security events, cases opened (incidents and vulnerabilities), metrics with security statistic about the system's devices and services. It provides a green/yellow/red indicator displaying the attacks and/or malicious activity over the IT devices/services.
Input data: A list of the devices and their security events (attacks and/or malicious activity, which may or not be prevented).
Output: A visual display of the IT devices/services' security status.
Suggested frequency: Daily.

2.3 Technology

2.3.1 EPS - Events per second [8]

Definition: The EPS metric calculates the average EPS collected into the SIEM. This metric helps to monitor the performance of all SIEM infrastructure's components aiding in the detection of overload and unresponsive components.
Input data: EPS for each SIEM device and collector.
Output: Average (daily) EPS for each SIEM device, collector, and for all the SIEM infrastructure.
Suggested frequency: Daily.

2.3.1.1 PE - Peak Event [8]

Definition: The PE metric calculates the average of the peak event (PE) for each SIEM device, collector and overall. The PE metric grants a quantifiable information about the performance of the devices in the presence of extreme conditions. By adding the Peak Event of each device, or the Peak Event of each collector, the security manager will get an overall PE perspective.

Input data: Events flux of each device in the presence of extreme conditions. To have a more accurate average value is necessary an input data of a minimum period of 90 days.
Output: maximum number of events per second in an extreme condition, for each SIEM device, collector, and for the SIEM itself.
Suggested frequency: Daily.

2.3.1.2 NE - Normal Event [8]

Definition: The NE metric calculates the normal behaviour of each SIEM device, collector and the SIEM itself, in the perspective of receiving events. The NE metric offers a quantifiable information about the performance of the devices in the presence of normal activity. By adding the Normal Event of each device, or the Normal Event of each collector, the security manager will get an overall NE perspective.
Input data: Events flow of each device and collectors in the presence of normal activity. To have a more accurate average value is necessary an input data of a minimum period of 90 days.
Output: The number of events per second in a normal state of operation, for each SIEM device, collector, and then for the SIEM itself.
Suggested frequency: Daily.

2.3.1.3 CELV - Changes of the event log volume [6]

Definition: This metric determines if a device is sending an abnormal number of events, providing to the team a faster response in detection and resolving the problems related with the device, connector or the communication between the two. It uses the three metrics above (EPS, PE and NE) to identify those devices.
Input data: Results of the EPS, PE and NE metrics.
Output: Device name which is having an abnormal number of events.
Suggested frequency: Daily.

2.3.2 TE - Top events [1]

Definition: The TE metric determines the most severe events received by the SIEM. It helps the team to detect the severe events, their types and the source which is providing them, and helps to analyse if the priority formula in the SIEM is classifying the events correctly.
Input data: events and their severity classified by the SIEM.
Output: a list of the most severe events received by the SIEM.
Suggested frequency: Daily.

2.3.3 PAM - Percentage of assets modelled (derived from [1])

Definition: The PAM metric calculates the percentage of assets being tracked by the SIEM (or other security technology), providing a view of the security team's monitoring surface.
Input data: All organization’s assets and the assets being tracked by the SIEM.
Output: Percentage of assets being tracked by the SIEM.
Suggested frequency: Weekly and/or Monthly.

2.3.4 PDM - Percentage of devices monitored (derived from [1])

Definition: The PDM metric calculates the percentage of devices (or data feeds) being fed into the SIEM by type. It can be used to track the devices that are being used to feed the SIEM and from those identify which are not providing events, due, for example, bad configuration.
Input data: All the devices which should be feeding the SIEM and the archive of events.
Output: Percentage of the devices that are truly being used to feed the SIEM.
Suggested frequency: Monthly.

2.3.5 EM - Events Management (derived from [1])

Definition: The EM metric calculates the number of raw events, uncorrelated events, correlated events and annotated events managed within the SIEM infrastructure. With the combination of these values the security manager can extract two valuable information: 1) the importance and the performance of the SIEM in the organization; 2) the ability of the SIEM to reduce the volume of the raw events to uncorrelated events and then correlate those events. With this metric is possible to analyse, over time, if SIEM's performance is increasing or decaying. The second valuable information is to check whether the analysts are executing the proper follow-up of the cases, by annotating and associating with events of interest.
Input data: All the events managed within the SIEM.
Output: The total number of raw events, uncorrelated events, correlated events and the annotated events managed within the SIEM.
Suggested frequency: Monthly.

2.3.6 DTD - Detection to Decision [5]

Definition: The DTD metric calculates the time it takes for an event/activity to be detected and processed through the detection tools, SIEM infrastructure, etc., before it reaches to the analyst. To calculate the required time, the DTD metric uses the timestamps associated with the events.
Input data: Events with timestamps.
Output: Time taken for an event to be detected and processed before it reaches to the analyst. Extra: average DTD, minimum DTD and maximum DTD.
Suggested frequency: Monthly.

2.3.7 SEU - SIEM resource usage (derived from [4])

Definition: The SEU metric is an indicator of the amount of CPU, RAM and disk resources used by the SIEM. The security manager can create alerts when the values are too high (or too low).
Input data: SIEM's list resources usage.
Output: The amount of CPU, RAM and disk resources used by the SIEM.
Suggested frequency: Monthly.

2.3.8 RH - Rules handled (derived from [4])

Definition: The RH metric calculates the total number of rules handled by the analysts (and not being acknowledged), providing information about the rules capacity (rules fired vs alerts handled), and how the rules are processing the events.
Input data: The rules fired.

Output: The total number of rules fired vs the total number of rules handled by the analysts, etc.
Suggested frequency: Monthly.

2.3.9 ID/PA - Intrusion Detection / Prevention Activity [1]

Definition: The ID/PA calculates statistics related with the ID/IPS systems and their effectiveness in the intrusion detection.
Input data: List of events and incidents.
Output: The total number of attacks detected by priority and number of attacks blocked (IPS only).
Suggested frequency: Daily.

2.3.10 QF - Quiet Feeds [1]

Definition: The QF metric calculates the number of feeds which are not giving any information. This can occur due to an interruption or discontinuity of the information given to the feed. With this information, the manager can discard the useless feeds.
Input data: feeds and the information provided by each feed.
Output: the feeds or the number of feeds not giving information.
Suggested frequency: Monthly.

2.3.11 PL - Patch Latency [2]

Definition: The PL metric calculates the time between a patch's release and the successful deployment of that patch in the organization.
A patch discovery service should be used to obtain the criticality of each missing patch and to calculate the time between the missing patches were introduced and the date of the scan to determine how long each missing patch has been available for each device.
Input data: The result of the patch discovery scan.
Output: a list by patch criticality with the respective time which the organization was unpatched.
Suggested frequency: Monthly.

2.3.12 PS - Patch Status (derived from [1])

Definition: The PS metric calculates the percentage of the systems that have the latest patches (Operating Systems or application) installed. In time of an attack or imminent crysis, is useful for the organization to detect the systems that aren't with the latest patch for that vulnerability.
Input data: The result of the patch discovery scan.
Output: Percentage of the systems without latest patch.
Suggested frequency: Monthly

2.3.13 ACover - Antivirus Coverage (derived from [1][6])

Definition: The ACover metric computes the risk in the organization concerning the devices which do not have antivirus installed and/or the latest antivirus definition files.
Input data: The result of the scan to determine which devices have antivirus installed and/or the latest virus definition files and a list of all the devices in the organization.

Output: The percentage of the devices which do not have antivirus installed and/or the latest virus definition files.
Suggested frequency: Monthly.

2.3.14 AStatus - Antivirus Status (derived from [1][6]

Definition: The AStatus metric computes the risk in the organization, concerning the antivirus policies status, verifying which of the antivirus installed aren't with the latest released policies and signatures.
Input data: The latest released policies and signatures available and the result of the scan containing the antivirus installed, their policies and signatures.
Output: Percentage of the antivirus installed which do not have the latest configurations, policies and signatures.
Suggested frequency: Monthly.

2.3.15 Top unusual scans / probe activities by source [6]

Definition: The unusual scans/probe activities metric calculates, by source, the top 10 (by occurrence) classified unusual scans and/or probe activities. This metric can be used to select which sources should be added to the blacklist.
Input data: The events containing scans and probes.
Output: a list of the top ten sources and their unusual scans and/or probe activities.
Suggested frequency: Monthly.

2.3.16 DUAC - Devices with unauthorized or anomalous communications [6]

Definition: This metric computes a list of devices with unauthorized or anomalous communications. By displaying the devices with unauthorized or anomalous communications the SOC team can react more quickly and effectively, thus allowing for a high-quality monitoring over the devices. An improvement of this metric would consider the criticality of the devices.
Input data: A list of events
Output: The devices with unauthorized or anomalous communications.
Suggested frequency: Daily.

2.3.17 UCC - Unusual configuration changes made in the FW, VPN, WAP and Domain (derived from [6])

Definition: This metric determines the latest (five for instance) unusual configuration changes in the four type of security devices (FW, VPN, WAP and Domain). It improves the SOC team monitoring process regarding security changes.
Input data: The events containing those unusual configuration changes.
Output: The latest unusual configuration changes in the four security devices.
Suggested frequency: Continuously.

2.3.18 TDT - Top dropped traffic by DMZ and FW [6]

Definition: This metric provides two points of view. One is the list of traffic categories with the biggest number of communications dropped by the DMZ and FW. The other is to monitor if the DMZ and FW aren't dropping traffic which should be forwarded.
Input data: A list of events.
Output: Top 10 dropped traffic from DMZ and FW.
Suggested frequency: Daily.

3 References

[1] ArcSight. (2010). WhitePaper: Security Operations Metrics Definitions for Management and Operations Teams. *HP ArcSight*, *44*(0), 0–7.

[2] Berinato, S. (2008). A few good information security metrics. Retrieved October 14, 2016, from http://www.csoonline.com/article/2118152/metrics-budgets/a-few-good-information-security-metrics.html

[3] Cain, C. I., & Couture, E. (2011). Establishing a Security Metrics Program. GIAC Enterprises, 1-27.

[4] Chuvakin, A. (2014). On SIEM Tool and Operation Metrics. Retrieved October 14, 2016, from http://blogs.gartner.com/anton-chuvakin/2014/06/17/on-siem-tool-and-operation-metrics/

[5] Cornell, C. (2015). Five Metrics You Should Be Recording for Incident Response. Retrieved October 14, 2016, from https://swimlane.com/five-metrics-for-incident-response/

[6] Gordon, S. (2015). Operationalizing Information Security Putting the top 10 SIEM best Practices to Work. *Processes, Metrics,(Technologies Introduction!*

[7] Kotenko, I., Polubelova, O., Saenko, I., & Doynikova, E. (2013). The ontology of metrics for security evaluation and decision support in SIEM systems. *Proceedings - 2013 International Conference on Availability, Reliability and Security, ARES 2013*, 638–645. https://doi.org/10.1109/ARES.2013.84

[8] SANS. (2009). Benchmarking Security Information Event Management (SIEM). *Event London*, (February), 14. Retrieved from http://www.sans.org/reading_room/analysts_program/eventMgt_Feb09.pdf

Appendix B

Public Blacklists

Table B.1: Public Blacklists and their information

Blacklist's name	**URL Link**	**Obs.**
badips_cyrusauth	https://www.badips.com/get/list/cyrusauth/age=1d	
badips_squid	https://www.badips.com/get/list/squid/?age=1d	
security_research	http://security-research.dyndns.org/pub/botnet/ponmocup/ponmocup-finder/ponmocup-infected-domains-latest.txt	
alienvault	https://reputation.alienvault.com/reputation.data	
lists_blocklist_ssh	https://lists.blocklist.de/lists/ssh.txt	
badips_apache-overflows	https://www.badips.com/get/list/apache-overflows/?age=1d	
ci-badguys	http://cinsscore.com/list/ci-badguys.txt	
emergingthreats_compromised-ips	http://rules.emergingthreats.net/blockrules/compromised-ips.txt	
lists_blocklist_ircbot	https://lists.blocklist.de/lists/ircbot.txt	
badips_apache-dokuwiki	https://www.badips.com/get/list/apache-dokuwiki/?age=1d	
badips_apache-defensible	https://www.badips.com/get/list/apache-defensible/?age=1d	

Continued on next page

Table B.1 – *Continued from previous page*

Blacklist's name	**URL Link**	**Obs.**
badips_Php-url-fopen	https://www.badips.com/get/list/Php-url-fopen/?age=1d	
nothink_http	http://www.nothink.org/blacklist/blacklist_malware_http.txt	
badips_qmail-smtp	https://www.badips.com/get/list/qmail-smtp/?age=1d	
badips_apache-scriddies	https://www.badips.com/get/list/apache-scriddies/?age=1d	
badips_apache-noscript	https://www.badips.com/get/list/apache-noscript/?age=1d	
badips_pop3	https://www.badips.com/get/list/pop3/?age=1d	
badips_bruteforce	https://www.badips.com/get/list/bruteforce/?age=1d	
nothink_irc	http://www.nothink.org/blacklist/blacklist_malware_irc.txt	
badips_pureftpd	https://www.badips.com/get/list/pureftpd/?age=1d	
virustotal	https://www.virustotal.com/vtapi/v2/ip-address/report	Requires API Key
dshield	http://www.dshield.org/ipsascii.html?limit=10000	
badips_local-exim	https://www.badips.com/get/list/local-exim/?age=1d	
lists_blocklist_bots	https://lists.blocklist.de/lists/bots.txt	
badips_proxy	https://www.badips.com/get/list/proxy/?age=1d	
badips_php-cgi	https://www.badips.com/get/list/php-cgi/?age=1d	
lists_blocklist_imap	https://lists.blocklist.de/lists/imap.txt	
badips_drupal	https://www.badips.com/get/list/drupal/?age=1d	
badips_nginx	https://www.badips.com/get/list/nginx/?age=1d	

Continued on next page

Table B.1 – *Continued from previous page*

Blacklist's name	**URL Link**	**Obs.**
badips_dovecot-pop3	https://www.badips.com/get/list/dovecot-pop3/?age=1d	
badips_sql	https://www.badips.com/get/list/sql/?age=1d	
badips_unknown	https://www.badips.com/get/list/unknown/?age=1d	
badips_proftpd	https://www.badips.com/get/list/proftpd/?age=1d	
badips_sip	https://www.badips.com/get/list/sip/?age=1d	
badips_imap	https://www.badips.com/get/list/imap/?age=1d	
badips_http	https://www.badips.com/get/list/http?age=1d	
malc0de	http://malc0de.com/bl/IP_Blacklist.txt	
badips_ftp	https://www.badips.com/get/list/ftp/?age=1d	
badips_assp	https://www.badips.com/get/list/assp/?age=1d	
badips_vsftpd	https://www.badips.com/get/list/vsftpd/?age=1d	
lists_blocklist _bruteforcelogin	https://lists.blocklist.de/lists/bruteforcelogin.txt	
badips_apacheddos	https://www.badips.com/get/list/apacheddos/?age=1d	
badips_xmlrpc	https://www.badips.com/get/list/xmlrpc/?age=1d	
lists_blocklist _strongIP	https://lists.blocklist.de/lists/strongips.txt	
badips_postfix	https://www.badips.com/get/list/postfix/?age=1d	
badips_phpids	https://www.badips.com/get/list/phpids/?age=1d	
badips_wp	https://www.badips.com/get/list/wp/?age=1d	
lists_blocklist_ftp	https://lists.blocklist.de/lists/ftp.txt	
badips_sql-attack	https://www.badips.com/get/list/sql-attack/?age=1d	

Continued on next page

Table B.1 – *Continued from previous page*

Blacklist's name	**URL Link**	**Obs.**
nothink_ssh	http://www.nothink.org/blacklist/blacklist_ssh_day.txt	
badips_pureftp	https://www.badips.com/get/list/pureftp/?age=1d	
badips_courierauth	https://www.badips.com/get/list/courierauth/?age=1d	
badips_plesk-postfix	https://www.badips.com/get/list/plesk-postfix/?age=1d	
badips_vnc	https://www.badips.com/get/list/vnc/?age=1d	
badips_dns	https://www.badips.com/get/list/dns/?age=1d	
badips_exim	https://www.badips.com/get/list/exim/?age=1d	
badips_ssh	https://www.badips.com/get/list/ssh/?age=1d	
badips_wordpress	https://www.badips.com/get/list/wordpress/?age=1d	
zeustracker	https://zeustracker.abuse.ch/blocklist.php?download=badips	
badips_sasl	https://www.badips.com/get/list/sasl/?age=1d	
badips_apache-spamtrap	https://www.badips.com/get/list/apache-spamtrap/?age=1d	
badips_ssh-ddos	https://www.badips.com/get/list/ssh-ddos/?age=1d	
badips_rdp	https://www.badips.com/get/list/rdp/?age=1d	
dragonForce _VNCPROBE	https://dragonresearchgroup.org/insight/vncprobe.txt	
urlvir	http://www.urlvir.com/export-ip-addresses/	
badips_default	https://www.badips.com/get/list/default/?age=1d	
dragonForce_SSH	https://dragonresearchgroup.org/insight/sshpwauth.txt	
badips_ssh-blocklist	https://www.badips.com/get/list/ssh-blocklist/?age=1d	
badips_apache-wordpress	https://www.badips.com/get/list/apache-wordpress/?age=1d	

Continued on next page

Table B.1 – *Continued from previous page*

Blacklist's name	URL Link	Obs.
badips_nginxpost	https://www.badips.com/get/list/nginxpost/?age=1d	
badips_apache	https://www.badips.com/get/list/apache/?age=1d	
badips_apache-w00tw00t	https://www.badips.com/get/list/apache-w00tw00t/?age=1d	
badips_nginxproxy	https://www.badips.com/get/list/nginxproxy/?age=1d	
badips_sql-injection	https://www.badips.com/get/list/sql-injection/?age=1d	
badips_cms	https://www.badips.com/get/list/cms/?age=1d	
feodotracker	https://feodotracker.abuse.ch/blocklist/?download=ipblocklist	
lists_blocklist_apache	https://lists.blocklist.de/lists/apache.txt	
badips_w00t	https://www.badips.com/get/list/w00t/?age=1d	
badips_sshd	https://www.badips.com/get/list/sshd/?age=1d	
badips_ssh-auth	https://www.badips.com/get/list/ssh-auth/?age=1d	
badips_courierpop3	https://www.badips.com/get/list/courierpop3/?age=1d	
cryptophp_master	https://raw.githubusercontent.com/fox-it/cryptophp/master/ips.txt	
badips_smtp	https://www.badips.com/get/list/smtp/?age=1d	
badips_badbots	https://www.badips.com/get/list/badbots/?age=1d	
badips_apache-nohome	https://www.badips.com/get/list/apache-nohome/?age=1d	
danger_rulez	http://danger.rulez.sk/projects/bruteforceblocker/blist.php	
lists_blocklist_mail	https://lists.blocklist.de/lists/mail.txt	

Continued on next page

Table B.1 – *Continued from previous page*

Blacklist's name	**URL Link**	**Obs.**
emergingthreats _botcc	`http://rules.emergingthreats.net/blockrules/emerging-botcc.rules`	
turris_greylist	`https://www.turris.cz/greylist-data/greylist-latest.csv`	
badips_owncloud	`https://www.badips.com/get/list/owncloud/?age=1d`	
openbl	`https://www.openbl.org/lists/base_30days.txt`	
badips_username-notfound	`https://www.badips.com/get/list/username-notfound/?age=1d`	
IPList_IPset	`https://raw.githubusercontent.com/firehol/blocklist-ipsets/master/firehol_level1.netset`	
badips _screensharingd	`https://www.badips.com/get/list/screensharingd/?age=1d`	
malwaredomainlist	`http://www.malwaredomainlist.com/updatescsv.php`	
dragonForce _HTTP	`//dragonresearchgroup.org/insight/http-report.txt`	
ransomwaretracker	`https://ransomwaretracker.abuse.ch/feeds/csv`	
badips_spam	`https://www.badips.com/get/list/spam/?age=1d`	
labs_snort	`http://labs.snort.org/feeds/ip-filter.blf`	
badips_sshddos	`https://www.badips.com/get/list/sshddos/?age=1d`	
badips_ddos	`https://www.badips.com/get/list/ddos/?age=1d`	
cert	`http://www.cert.org/downloads/mxlist.ips.txt`	
cruzit	`http://www.cruzit.com/xwbl2csv.php`	
dips_apache-phpmyadmin	`https://www.badips.com/get/list/apache-phpmyadmin/?age=1d`	
badips_postfix-sasl	`https://www.badips.com/get/list/postfix-sasl/?age=1d`	

Continued on next page

Table B.1 – *Continued from previous page*

Blacklist's name	**URL Link**	**Obs.**
lists_blocklist_sip	https://lists.blocklist.de/lists/sip.txt	
badips_telnet	https://www.badips.com/get/list/telnet/?age=1d	
badips_dovecot-pop3imap	https://www.badips.com/get/list/dovecot-pop3imap/?age=1d	
badips_apache-php-url-fopen	https://www.badips.com/get/list/apache-php-url-fopen/?age=1d	
badips_apache-404	https://www.badips.com/get/list/apache-404/?age=1d	
badips_dovecot	https://www.badips.com/get/list/dovecot/?age=1d	
badips_asterisk	https://www.badips.com/get/list/asterisk/?age=1d	
badips_apache-modsec	https://www.badips.com/get/list/apache-modsec/?age=1d	
badips_named	https://www.badips.com/get/list/named/?age=1d	
badips_asterisk-sec	//www.badips.com/get/list/asterisk-sec/?age=1d	
osint	http://osint.bambenekconsulting.com/feeds/c2-ipmasterlist-high.txt	
autoshun	https://www.autoshun.org/download/?api_key=	Requires API KEY
badips_rfi-attack	https://www.badips.com/get/list/rfi-attack/?age=1d	
badips_spamdyke	https://www.badips.com/get/list/spamdyke/?age=1d	
sslbl	https://sslbl.abuse.ch/blacklist/sslipblacklist.csv	
charles	http://charles.thehaleys.org/ssh_dico_attack_hdeny_format.php/hostsdeny.txt	

Appendix C

UML for the framework solution

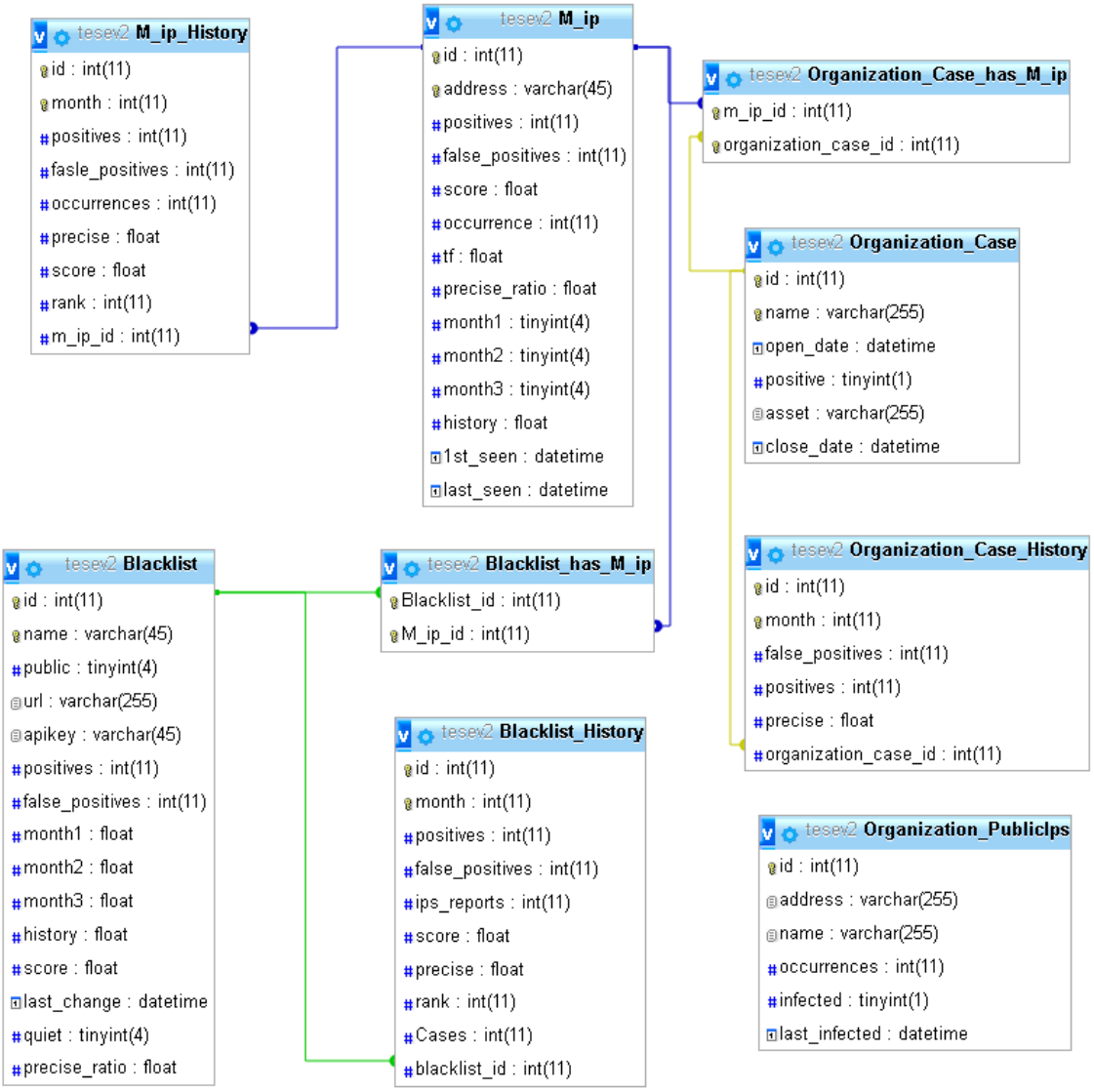


Figure C.1: Database UML